昭和58年6月18日発行(毎月1回18日発行)第2巻第7号通巻第14号　昭和58年6月7日国鉄首都特別扱承認雑誌第6897号

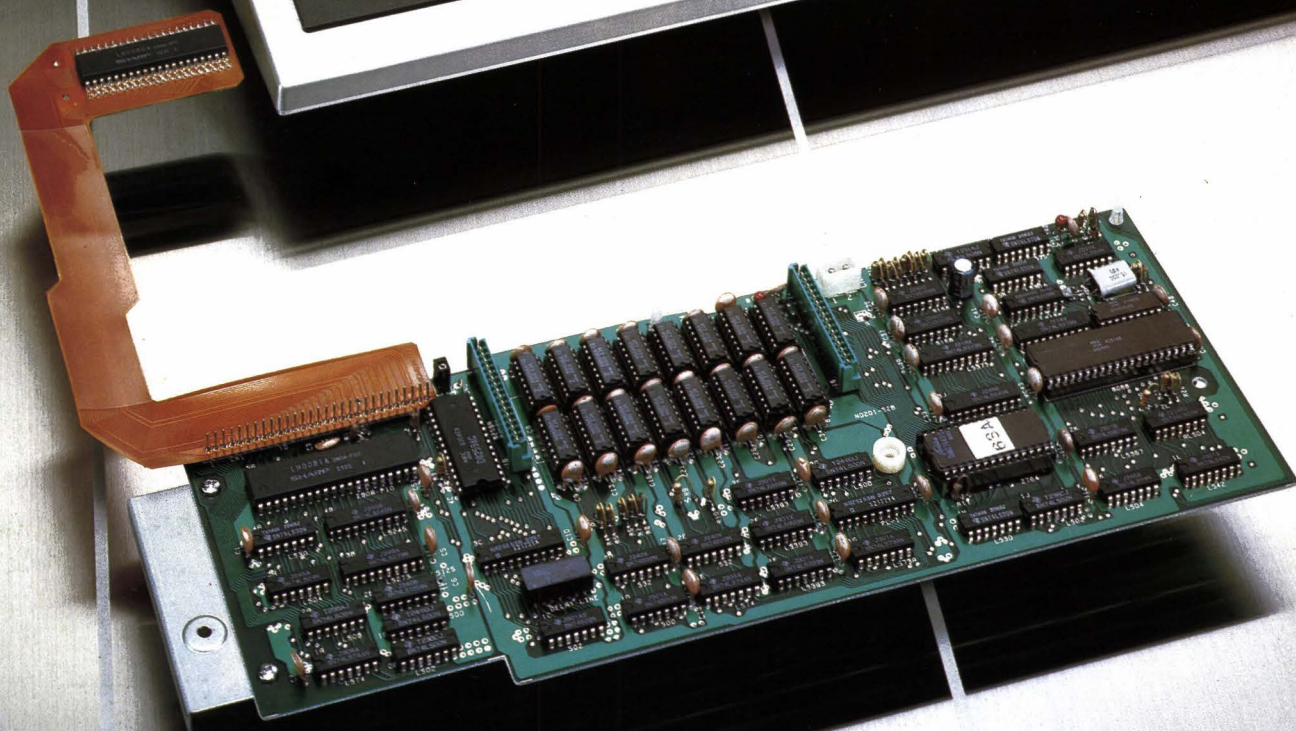
SHARP パソコン情報誌 Personal Computer Magazine 480円

Oh!MZ

MZ, X&ポケコンシリーズ

特集
マシン語プログラミング
逆アセンブラにハードコピーを
BASICの壁を越える
X1逆アセンブラと命令語の解析

●おもしろゲーム1等賞作品発表
X1 前線基地
●MZ-2000
実感16ビット体験
●MZ-2000／80B
Hu-BASIC V2.0の紹介

Jul. 1983 7

いま大いなる変身、MZ-2000。

16ビットボードキット、漢字ROMボード、カラーインクジェットプリンタによる強力なサポート。

クリーンコンピュータのベストセラー機MZ-2000に強力なサポート体制が整いました。16ビットマシンへの鮮やかな進化を実現する16ビットボードキット、加えて漢字化への対応が図れる漢字ROMボード、さらにカラーアウトプット時代をひらく新鋭機カラーインクジェットプリンタの登場で、MZ-2000はまさに究極の次元へと飛躍します。大いなる変身、この極限まで強化された機能をどう使いこなすか、いまMZはいよいよ興味深いコンピュータシーンを展開しようとしています。

16ビットマシンへの飛躍

16ビットボードキットMZ-1M01は、メインCPUとしてμPD8088を、サブCPUとしてZ80Aを持つ先進の16ビットコンピュータとしてMZ-2000を鮮やかに変身させます。しかもこの拡張されたMZ-2000は、LOADするシステムソフトウェアによって、単に16ビットマシンとして動作するだけでなく、8ビットマシンとして動作させることもでき、従来のMZ-2000の蓄積されたソフトウェアがそのまま使用できるなど、幅広いコンピュータライフが楽しめる設計です。16ビットマシンの機能としては、128Kバイト RAM実装によるより高度なプログラミング、カラー及び漢字処理、省力化されたキー操作など、新しい拡張機能がふんだんに盛り込まれています。

●CPU：μPD8088、Z80A●RAM：128Kバイト標準実装●カラー対応のBASIC[16]インタープリタ装備(カラー機能使用時にはオプションのグラフィックボードMZ-1R01及びグラフィックメモリMZ-1R02(×2)が必要です。)●8ビットモード/16ビットモード使い分け可能(8ビットモードでは従来のMZ-2000と完全コンパチ)●オートリピート、特殊コントロールキーなど省力化された便利なキー操作●漢字2965字、非漢字453字の漢字ROMボード(MZ-1R08 標準価格29,000円)装着可能(漢字表示にはオプションのグラフィックボードMZ-1R01が必要です。)

鮮やかな7色カラープリント

シャープ先進のエレクトロニクス技術が、ついに7色カラープリントができる低騒音インクジェットプリンタを完成、いよいよプリンタもカラー時代に突入しました。複雑なグラフ、図形から精緻なコンピュータアニメやグラフィックまで色鮮やかにプリント。しかもデスクサイドで手軽に使えるコンパクト設計。ビジネスはもちろん、教育、ホビーなど応用範囲も実に多彩、さまざまな分野で活用できます。いま、時代を告げる待望の新鋭機MZ-1P04の登場でMZ-2000は大きな飛躍を遂げたことになります。

16ビットボードキット **新発売**
MZ-1M01
標準価格78,000円

カラーインクジェットプリンタ
MZ-1P04
システム 標準価格248,000円
〈プリンタインターフェイスカード・接続ケーブル含む〉

パーソナルコンピュータ **MZ-2000**

パソコンに求められるあらゆる機能を搭載したMZの最上位バージョン

パーソナルコンピュータ
MZ-3500 シリーズ

MZ-3531 標準価格320,000円〈ミニフロッピー1基内蔵〉
MZ-3541 標準価格410,000円〈ミニフロッピー2基内蔵〉

●写真は本体(MZ-3541)、キーボード(MZ-1K06標準価格38,000円)、CRT(MZ-1D03標準価格163,000円)を組合せた例です。※画面はオプションのグラフィックボード、グラフィックメモリ(×2)を使用した例です。

大いなる発展性を秘めて、驚異のハイコストパフォーマンスMZ

パーソナルコンピュータ
MZ-700 シリーズ

MZ-711……標準価格 79,800円
MZ-721……標準価格 89,800円(データレコーダ内蔵)
MZ-731……標準価格128,000円(データレコーダ・カラープロッタプリンタ内蔵)

●写真はMZ-731とCRT(MZ-1D05標準価格69,800円)を組合せた例です。

話題沸騰!! TV提供番組"パソコン・サンデー"毎週日曜、以下の放送局で絶賛放映中!!

●テレビ大阪9:30～10:00●テレビ東京9:30～10:00●奈良テレビ12:00～12:30●テレビ和歌山9:30～10:00●びわ湖放送11:25～11:55
●札幌テレビ24:25～24:55●福島放送23:20～23:50●東北放送24:00～24:30●新潟放送24:00～24:30●信越放送24:00～24:30
●石川テレビ24:35～25:05●東海テレビ24:35～25:05●山陽放送24:05～24:35●広島テレビ7:00～7:30●テレビ西日本24:36～25:06
●山梨放送7:30～8:00●テレビ静岡24:35～25:05●熊本県民テレビ8:30～9:00●琉球放送24:00～24:30

シャープ株式会社 本社 〒545 大阪市阿倍野区長池町22番22号 ☎(06)621-1221(大代表)●お問い合わせは…本社内国内家電営業本部システム機器営業企画部またはシャープエンジニアリング㈱およびシャープ㈱国内産機営業本部へ。詳しくは後の情報ページをご参照ください。

SHARP

充実したサポート体制で幅広い拡張性を誇る

●MZ-2000・80B2 システム構成図

クリーンコンピュータ
MZ-80B2
(グラフィックRAM付)
標準価格 278,000円

本体内に収納

拡張I/Oポート
MZ-8BK
標準価格
19,800円

パーソナルコンピュータ
MZ-2000

拡張ユニット
MZ-1U01
標準価格
37,000円

グラフィック表示用
グラフィックボード(ページ1)
MZ-1R01

グラフィックメモリ
MZ-1R02
(ページ2またはページ3)

14型カラーディスプレイ
MZ-1D01
標準価格 124,000円
※カラーディスプレイをコントロールするには拡張カラーBASICが必要です。

12型カラーディスプレイ
MZ-1D06
標準価格 102,000円

プリンタ用I/Oカード
MZ-8BP5I
標準価格
17,400円

接続ケーブル
MZ-8BP4C
標準価格 11,000円

ドットプリンタ
MZ-80P4B
標準価格
281,000円

接続ケーブル
MZ-8BP5C
標準価格 8,600円

ドットプリンタ
MZ-80BP5
標準価格
142,000円

ドットプリンタ
MZ-80P6
標準価格
155,000円

ユニバーサルI/Oカード
MZ-8BI01
標準価格 16,700円

ユニバーサルI/Oコネクタ
MZ-8BCN
標準価格 2,700円

PTR・PTP等

RS-232C用ケーブル
MZ-8BC03
標準価格 11,000円

RS-232Cシリアルインターフェイスカード
MZ-8BI03
標準価格 50,000円

GP-IB用ケーブル
MZ-8BC04
標準価格 18,000円

モデム・テレタイプ等

GP-IBインターフェイスカード
MZ-8BI04
標準価格
45,000円

各種計測器

RS-322C・GP-IB用テープBASIC
MZ-8BT06
標準価格 9,500円

MZ-80B
RS-232C・GP-IB用ディスクベーシック
MZ-8BD03
標準価格 12,000円

プリンタインターフェイス
MZ-1E04
標準価格 11,000円

接続ケーブル
MZ-1C24
標準価格 9,000円

カラーインクジェットプリンタ
MZ-1P04
標準価格 228,000円
(インターフェイス、接続ケーブルを含むセット標準価格 248,000円)

フロッピー用I/Oカード
MZ-8BFI
標準価格 38,000円

接続ケーブル
MZ-8BFC
標準価格 8,700円

フロッピーディスクドライブ
MZ-80BF

拡張用接続ケーブル
MZ-8BFKC
標準価格 8,400円

拡張用フロッピーディスクドライブ
MZ-80BFK

接続ケーブル
MZ-8KFC
標準価格
8,700円

シングルフロッピーディスクドライブ
MZ-80SFD
標準価格
158,000円

拡張用接続ケーブル
MZ-80F05
標準価格 3,700円

シングルフロッピーディスクドライブ
MZ-80SFD
標準価格
158,000円
(4ドライブまで接続可能)

MZ-2000マスターディスク
MZ-2Z001
標準価格
10,000円

MZ-80Bマスターディスク
MZ-8BDM
標準価格 10,000円

マークカード用I/Oセット
MZ-80MCB
標準価格 30,000円

マークカードリーダ
MZ-80MCR
標準価格 198,000円

カラーディスプレイ用I/Oセット
MZ-8DUBI
標準価格 32,000円

インテリジェントカラーターミナル
MZ-80DUA

カラーモニタ
MZ-80DUB

14型カラーディスプレイユニットMZ-80DU
標準価格 262,000円

グラフィック表示用
拡張グラフィックRAMII
MZ-8BGK
標準価格 39,000円

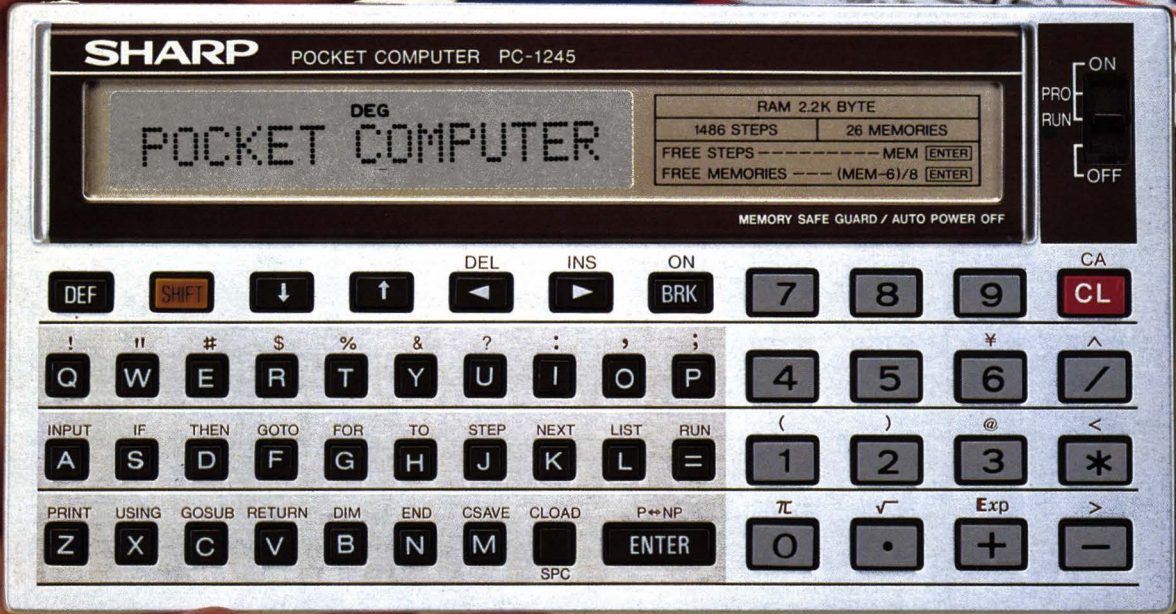
●MZ-700シリーズシステム構成図

MZクリーンコンピュータ。

MZ-80B2システムソフトウェア

品名	型名		価格
●フロッピーDOS(BASICコンパイラ含む)	MZ-8BD02	標準価格	50,000円
●倍精度BASIC(ディスク)	MZ-8BD01	標準価格	10,000円
●倍精度BASIC(テープ)	MZ-8BT03	標準価格	5,000円
●マシンランゲージ(テープ)	MZ-8BT05	標準価格	7,000円
●インタープリタPASCAL(テープ)	MZ-8BT02	標準価格	10,000円
●システムプログラム(テープ)	MZ-8BT04	標準価格	25,000円
●RS-232C・GP-IB用ディスクBASIC	MZ-8BD03	標準価格	12,000円
●RS-232C・GP-IB用テープBASIC	MZ-8BT06	標準価格	9,500円
●BASICプログラミングガイド	MZ-8BM01	標準価格	3,000円
●BASICコンバートテープ	MZ-8BT01	標準価格	2,700円

MZ-2000システムソフトウェア

品名	型名		価格
●カラーテープBASIC	MZ-1Z002	標準価格	7,000円
●倍精度テープBASIC	MZ-1Z003	標準価格	7,000円
●インタープリタPASCAL	MZ-1Z004	標準価格	12,000円
●システムプログラム	MZ-1Z005	標準価格	25,000円
●マシンランゲージ	MZ-1Z006	標準価格	7,000円
●RS-232C・GP-IBコントロールBASIC	MZ-1Z010	標準価格	9,500円
●カラーディスクBASIC	MZ-2Z002	標準価格	12,000円
●倍精度ディスクBASIC	MZ-2Z003	標準価格	12,000円
●フロッピーDOS	MZ-2Z004	標準価格	50,000円

いつのまにか身についた。

うれしい手帳サイズ、独習テキストもついてくる。マスターすれば即実務に応用できる高機能設計。

いつでも、どこでも、手軽に使えるコンピュータ、シャープのポケットコンピュータに新機種PC-1245が登場しました。ワイシャツのポケットに納まる手帳サイズ、しかもこのコンパクトなボディに強力BASICを搭載。優れた処理能力とともに、楽しく学習できる入門書をはじめ、BASICを意識したうれしい配慮。これからBASICを学びたい、BASICを活用して実務にいかしたい、そうした人々にぜひおすすめしたいポケットコンピュータの自信作です。

BASICのためのワンタッチコマンド入力

PRINT、INPUTなど使用頻度の高い18種類のBASIC命令語が簡単に入力できる便利機能。プログラムがスピーディに作成できます。

高速C-MOS8ビットCPU搭載、多彩に活用できる大容量メモリー

システムの頭脳部CPUには、シャープ独自の開発による高速C-MOS8ビットCPUを採用。複雑な事務計算、技術計算など時間を要する計算もスピーディに処理。しかもC-MOSだから低消費電力です。またメモリー容量はコンパクトながらROM24Kバイト、RAM2.2Kバイトを標準実装。ビジネスをはじめ、さまざまな分野で多彩に活用できる能力を誇っています。

実用的な16桁表示、操作性に優れたキーボード

表示部は余裕の16桁、科学技術計算など高精度を必要とする10桁の指数が完全に表示できる他、プログラム表示、答え表示など対話しながらのオペレーションが可能です。またキーボードは、使用頻度の高い数字キー、ファンクションキーを大きくするなど、手帳サイズながら操作性にはキメ細かな配慮がなされています。さらにアルファベットキーのうち18個は、プログラムの頭出しができる定義付キーとして使えます。

充実のコマンドを誇る強力BASIC

優れたプログラム機能、文字処理機能によって、入門から実務まで幅広く使いこなせるBASICです。たとえば2次元配列変数を使えば、分類集計などもスムーズに行えますし、文字の並べかえなど複雑な処理も文字関数により簡単にプログラムできます。

楽しみながらBASICがマスターできる入門書

コンピュータを初めて扱う人にもわかりやすく、親しみやすいよう、例やイラストをふんだんに盛り込み、読んでいるうちに自然とBASICの基礎がマスターできるよう工夫されています。

多彩な付加機能

●プログラムが消えないメモリー保護機能●プログラムのリスト操作を実行させないパスワード機能●アラームや効果音として使えるBEEP音機能●プログラム入力を簡単にする省略形入力

●外部記憶装置としてのテープレコーダが接続できるカセットインターフェイスCE-124 標準価格 4,500円

新発売

ポケットコンピュータ PC-1245

標準価格 17,800円

●寸法/重量:幅135×奥行70×厚さ9.5㎜/115g

手帳サイズのコンピュータ、オールインワンシステムにもなってA5判サイズ。

ポケットコンピュータ PC-1251/1250

PC-1251 標準価格29,800円
PC-1250 標準価格22,800円

●寸法:幅135×奥行70×厚さ9.5㎜●重量:115g(電池含む)●オプション:一瞬にしてオールインワンシステムが組めるプリンタ/マイクロカセットレコーダCE-125 標準価格 36,800円◀写真はPC-1251+CE-125

ミニグラフィック表示、カラーグラフィック印字など驚異の性能を誇る本格派。

ポケットコンピュータ PC-1500

標準価格 59,800円

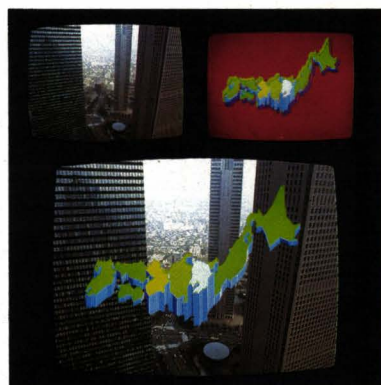
▲写真はPC-1500+CE-150●4色で様々なパターンのグラフや図形が描けるカラーグラフィックプリンタCE-150 標準価格 49,800円

SHARP
SHARP
POWER
EJECT
CENTRAL PROCESSING UNIT
SHARP
X1
KEY BOARD
SHARP
BREAK
SHIFT
ESC
H TAB
CTRL
SHIFT
GRAPH
CAPS LOCK
X1

プロも、まっ青。スーパーインポーズ。

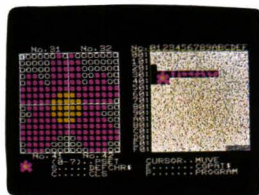
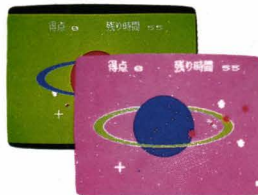
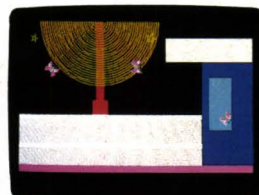
コンピュータ画像とテレビ映像との自在なクロスオーバー、シャープ先進エレクトロニクスがひらく21世紀のメディア。

●いま未体験ゾーンへの突入。パーソナルコンピュータの世界に新しいメディアをひらくパソコンテレビX1──。シャープは先端エレクトロニクス技術でパソコンとテレビのシステム化を実現。テレビ・ビデオ映像とコンピュータとのクロスオーバー、世界初のスーパーインポーズ機能をはじめとした新鮮なグラフィックスは、来たるべき映像情報化時代に代表される新しいメディアの核として熱い視線を浴びています。●もちろん、パーソナルコンピュータとしての高機能も見逃がせない特長です。3CPU構成、伝統のクリーン思想を受け継いだメインメモリ64Kバイト、オールRAM。自動頭出し機能を備え、ディスクに迫る操作性を誇る2700ボーの高速カセットレコーダ内蔵。そしてこうしたハードウェアに密接に結びついてその機能を最大限に発揮させる強力なソフトウェア……パソコンテレビX1は、究極の8ビットマシンとして、ホームユースからビジネスユースまで幅広い分野への応用が可能な、まさに21世紀のメディアです。

●メインCPUにZ80A(4MHz)、周辺コントロール用として2個のサブCPU搭載●RAM：メインメモリ64Kバイト、テキスト用V-RAM4Kバイト、ユーザー定義キャラクターゼネレータ6Kバイト、グラフィック用V-RAM(48Kバイト・オプション)●カラーBASICインタプリタ装備●セントロニクスプリンタインターフェイス、ジョイスティックインターフェイス内蔵●8オクターブ3和音のサウンドゼネレータ●日付、時刻、番組予約が可能なカレンダー付タイマー●小さな文字や細かな縞模様もくっきりと再現する新開発ファインピッチブラウン管●RGB接続端子とビデオ入出力端子付

●スーパーインポーズ機能

新開発のASC(オートマチック・シンクロナイズ・コントロール)回路により、コンピュータ信号とテレビ信号のスーパーインポーズを実現しました。テレビ画面にコンピュータ画面を重ね合わせて自由に表示することができます。

●スーパーインポーズ機能で作成した重ね合わせ画像をビデオ録画できるデジタルテロッパー

デジタルテロッパーCZ-8DT　標準価格89,800円

キャラクターゼネレータ

パレット機能

プライオリティ機能

〈パソコンテレビX1が誇る強力なグラフィック機能〉●256種類のキャラクタが自由に定義できるプログラマブル・キャラクタ・ゼネレータ●多画面切換えができる高機能グラフィック●図形や文字の色を瞬時に変えられるパレット機能●遠近感、立体感のある画面がつくれるプライオリティ機能●グラフィックの拡大縮小、移動ができる座標変換機能●任意の面の色づけが簡単にできるタイリングペイント機能●大小の文字を同一画面に混在できるアトリビュート

パソコンテレビX1

パーソナルコンピュータ+キーボードCZ-800C 標準価格155,000円・14型カラーディスプレイテレビCZ-800D 標準価格113,000円

●別売周辺機器＝グラフィックRAM CZ-8GR(標準価格32,000円)/拡張I/OポートCZ-8EP(標準価格11,800円)/フロッピーディスクCZ-800F(標準価格198,000円)/フロッピーディスクインターフェイスCZ-8FA(標準価格24,000円)/ドットプリンタCZ-800P(標準価格142,800円)/漢字ROM CZ-8KR(標準価格38,000円)/RFビデオコンバータCZ-8VC(標準価格15,800円)

●写真はパーソナルコンピュータ+キーボード(CZ-800C)と14型カラーディスプレイテレビ(CZ-800D)を組合せた例です。なおスーパーインポーズはこのシステムで初めて作成可能。●画面は別売のグラフィックRAMを使用し作成、印刷合成したものです。▶パソコンテレビX1には、ローズレッド・メタリックシルバー・スノーホワイトの3色があります。

シャープ株式会社 本社 〒545 大阪市阿倍野区長池町22番22号 ☎(06)621-1221(大代表)●お問い合わせは…本社内電子機器事業本部システム機器営業部・国内家電営業本部システム機器営業企画部またはシャープエンジニアリング㈱およびシャープ㈱国内産機営業本部へ。詳しくは後の情報ページをご参照ください。

資料請求券 X1 oh!MZ 7係

SHARP

「安心」もごいっしょに。全国をカバーする

シャープエンジニアリング株式会社（MZシリーズ、パソコンテレビX1へのお問い合わせは下記へどうぞ。）

北海道地区

シャープエンジニアリング㈱ 札　幌
〒063 札幌市西区24軒1条7丁目3-17
☎(011)642-4649

札　幌SS (011)641−4649
北　見SS (0157)36−4649
遠　軽SB (01584)2−1137
帯　広SB (0155)25−6832
苫小牧SB (0144)32−4649
室　蘭SB (0143)45−4649
岩見沢SS (01262)4−4649
滝　川SB (0125)22−0200
釧　路SS (0154)25−4649
根　室SB (01532)4−4800
旭　川SS (0166)25−4649
稚　内SB (01622)2−4764
道　南SS (0138)51−4649

東北地区

シャープエンジニアリング㈱ 仙　台
〒983仙台市六丁目字本新田東1
☎(0222)88-9141

宮　城SS (0222)96−4649
仙　南SB (02245)3−4649
北宮城SS (02292)2−5520
石　巻SB (0225)96−5627
気仙沼SB (0226)23−1588
青　森SS (0177)38−0281
弘　前SB (0172)36−6425
む　つSB (01752)2−7380
八　戸SS (0178)44−4649
十和田SB (01762)2−4649
岩　手SS (0196)38−6087
釜　石SB (0193)23−4649
水　沢SB (01972)3−8428
宮　古SB (01936)3−5658
秋　田SS (0188)63−4649
大　館SB (0186)49−2975
羽　後SS (01823)3−4649
本　荘SB (01842)3−4649
山　形SS (0236)31−4649
酒　田SB (0234)24−4649
新　庄SB (02332)3−1277
福　島SS (0249)45−4649
会津若松SB (0242)25−4649
福　島SB (0245)53−4649
いわきSS (0246)28−4649
原　町SB (02442)2−5025

関越地区

シャープエンジニアリング㈱ 宇都宮
〒320 宇都宮市不動前4丁目2番41号
☎(0286)35-1151

栃　木SS (0286)37−1178
新発田SB (02542)4−2646
小　山SB (0285)22−4649
群　馬SS (0272)52−4649
太　田SB (0276)45−3241
茨　城SS (0292)41−4649
南茨城SS (0298)22−6111
新　潟SS (0252)85−4649
西那須野SB (02873)6−7131
三　条SB (02563)8−6761
上　越SB (0255)23−7148
長　岡SS (0258)35−8254
松　本SS (0263)25−7536
飯　田SB (0265)24−0640
岡　谷SB (02662)3−8421
長　野SS (0262)28−4649
上　田SB (0268)27−1329

東京地区

シャープエンジニアリング㈱ 東　京
〒114 東京都北区東田端2丁目13番17号
☎(03)893-4649

江　東SS (03) 626−4649
城　東SS (03) 629−4649
城　南SS (03) 776−4649
城　西SS (03) 382−4649
城　北SS (03) 972−4649
三多摩SS (0425)84−4649
武蔵野SS (0422)32−4649
埼　玉SS (0486)66−4649
埼玉南SS (0484)45−6996
埼玉北SB (0485)53−3111
埼玉東SB (0487)61−3511
埼玉西SB (0492)46−1655
千　葉SS (0472)65−4649
館　山SB (04702)2−3227
西千葉SS (0473)68−4649
船　橋SB (0474)24−8003
銚　子SB (0479)23−3373
横　浜SS (045)753−4649
川　崎SS (03) 735−4649
横須賀SB (0468)36−9883
湘　南SS (0463)54−4649
小田原SB (0465)23−0271
相模原SB (0462)75−8790
山　梨SS (0552)26−4649
静　岡SS (0542)85−4649
沼　津SS (0559)22−4649

北陸地区

シャープエンジニアリング㈱ 金　沢
〒921 石川県石川郡野々市町御経塚1096の1
☎(0762)49-4649

石　川SS (0762)49−4649
七　尾SB (07675)3−4649
小　松SB (0761)22−4649
富　山SS (0764)51−4649
高　岡SS (0766)25−4649
福　井SS (0776)54−4649
敦　賀SB (07702)3−4649

中部地区

シャープエンジニアリング㈱ 名古屋
〒454 名古屋市中川区山王3丁目5番5号
☎(052)322-4649

名古屋SS (052)332−2621
北名古屋SS (0587)66−4649
半　田SS (0569)22−4649
岡　崎SS (0564)24−4649
豊　橋SS (0532)53−4649
浜　松SS (0534)63−4649
三　重SS (0592)32−6200
伊　勢SB (0596)36−1100
四日市SS (0593)51−4649
伊賀上野SB (0595)21−2228
岐　阜SS (0582)73−4649
大　垣SB (0584)89−5771
濃　飛SS (05742)6−4649
高　山SB (0577)33−6761

(SS…サービスステーション
SB……サービスブランチ)

近畿地区

シャープエンジニアリング㈱ 大　阪
〒556 大阪市浪速区恵美須西1丁目2番9号
☎(06)643-4649

大　阪SS (06) 643−4649
東大阪SB (0729)94−4649
阿倍野SB (06) 629−1741
北大阪SS (06) 328−4649
南大阪SS (0722)45−4649
岸和田SS (0724)44−4649
神　戸SS (078)453−4649
明　石SB (078)927−7404
淡　路SB (07992)4−2917
阪　神SS (06) 421−4649
姫　路SS (0792)66−1818
豊　岡SB (07962)3−7389
滋　賀SS (0775)25−7856
彦　根SB (0749)22−3299
京　都SS (075)672−2375
舞　鶴SB (0773)75−0653
奈　良SS (07435)3−6691
奈良南SB (07456)5−1491
和歌山SS (0734)45−4649
南　紀SS (0739)25−3011
新　宮SB (0735)22−0900

中国地区

シャープエンジニアリング㈱ 広　島
〒731-01 広島市安佐南区祇園町大字西原2249の1
☎(082)874-4649

広　島SS (082)874−2281
東広島SB (0824)28−4649
福　山SS (0849)51−4649
岡　山SS (0862)41−4649
津　山SB (08682)2−6296
倉　敷SB (0864)22−2183
山　口SS (0834)31−4155-6
下　関SB (0832)53−1065
山口中央SB (08397)2−7318
山　陰SS (0852)24−4649
浜　田SB (08552)2−1521
出　雲SB (0853)22−4649
鳥　取SB (0857)22−8278
米　子SB (0859)29−7311

四国地区

シャープエンジニアリング㈱ 高　松
〒760 高松市木太町1861の3
☎(0878)33-4649

香　川SS (0878)33−4649
徳　島SS (0886)25−4649
高　知SS (0888)82−4649
中　村SB (08803)5−2138
愛　媛SS (0899)71−4649
東　予SS (0897)41−8840
南　予SS (0895)25−4649

九州地区

シャープエンジニアリング㈱ 福　岡
〒816 福岡市博多区井相田2丁目12番地の1
☎(092)572-4649

福　岡SS (092)572−4649
佐　賀SB (0952)24−9462
久留米SS (0942)21−1251
北九州SS (093)592−5961-2
筑　豊SS (09482)3−7519
大　分SS (0975)36−3909
長　崎SS (0958)44−4649
佐世保SB (0956)32−6666
熊　本SS (0963)66−4649
大牟田SB (0944)55−5111
八　代SB (09653)2−2188
天　草SB (09692)3−8711
鹿児島SS (0992)53−4649
川　内SB (0996)25−3777
宮　崎SS (0985)24−6723
延　岡SB (0982)34−5735
都　城SB (0986)24−2235

沖縄地区

沖縄シャープ電機
〒900 沖縄県那覇市曙2丁目10-1
☎(0988)62-2231

沖　縄SS (0988)62−2231
宮　古SB (09807)2−3436
石　垣SB (09808)2−4072
沖縄中部SS (09893)7−9912
沖縄北部SB (09805)2−1506
奄　美SB (09975)3−4777

●お気軽にどうぞ、MZマイコンセミナー

北海道地区

STV文化教室
☎(011)642-7156
● BASIC入門A・Bコース

朝日カルチャーセンター
☎(011)222-4121
● BASIC入門コース

道新文化センター
☎(011)221-2111
● マイコン初級コース

道新文化センター〈苫小牧〉
☎(0144)33-5350
● マイコン初級コース

道新文化センター〈小樽〉
☎(0134)23-1700
● マイコン初級コース

4プラ・コンピュータ教室
☎(011)261-0221
● レディスコース ● OLコース
● ママさんコース
● ビジネスマンコース
● お子様コース ● 熟年コース

旭川マイコン教室
☎(0166)33-3300
● BASIC入門・初級・中級・上級・ビジネスコース

東北地区

FTV学苑〈福島TV〉
☎(0245)21-3232
● BASIC初級・演習コース

FCT学園
☎(0249)23-5564
● BASIC入門コース

FBC福島文化学園マイコン教室
☎(0245)22-9118
● BASIC初級・中級コース

東北電子計算機専門学校
☎(0222)22-8931
● BASIC入門コース

仙台放送教養学園
☎(0222)63-0191
● BASIC入門・中級コース

関越地区

栃木マイコン学院
☎(0286)37-3997
● マイコン入門・初級コース

リリーコミニティーカレッジ
☎(0292)21-9476
● マイコン基礎・応用コース

足利コンピュータ学院
☎(0284)72-2981
● BASIC入門コース
● ビジネスマンコース

東京地区

静岡けんみんテレビ
☎(0542)51-3300
● BASIC入門コース

朝日カルチャーセンター〈新宿〉
☎(03)344-1941
● BASIC入門コース

ダイヤモンドビジネス
☎(03)504-6476
● ビジネスマンコース
● 経営者・管理者コース

山梨文化学園
☎(0552)53-1111
● BASIC入門コース

シャープマイコン教室
☎(03)260-1161
● BASIC入門コース

コンピュータプラザ・ニデコ
☎(03)251-8061
● BASIC入門講座

パソコンクリニック
☎(03)234-4141
● ビジネス入門コース
● 管理者コース
● 業務別・業種別コース

北陸地区

北国文化センター
☎(0762)22-0101
● BASIC入門コース

能森学院富山マイコンスクール
☎(0764)32-2513
● BASIC入門・中級コース

金沢マイコンスクール
☎(0762)47-5107
● 初級・中級コース

富山テレビパソコンスクール
☎(0764)25-1111
● 1日入門・初級・中級コース
● 応用・マンツーマンコース

信頼の情報ネットワーク。

シャープビジネス株式会社（MZシリーズ、パソコンテレビＸ１、ポケットコンピュータへのお問い合わせは下記へどうぞ）

北海道地区

シャープビジネス㈱札幌
〒063 札幌市西区24軒1条7丁目3番17号
☎(011)641-3631

- 函館 (0138)55−0232
- 旭川 (0166)25−2585
- 釧路 (0154)22−9777
- 帯広 (0155)35−4365
- 北見 (0157)36−6848

東北地区

シャープビジネス㈱仙台
〒983 仙台市六丁目字本新田東1
☎(0222)88-9151

- 郡山 (0249)45−4380
- いわき (0246)28−1181
- 青森 (0177)38−7501
- 八戸 (0178)44−7713
- 盛岡 (0196)38−8868
- 秋田 (0188)62−0027
- 山形 (0236)33−3215
- 庄内 (0234)24−5344

関越地区

シャープビジネス㈱新潟
〒950 新潟市上所中1丁目7番21号
☎(0252)83-1795

- 長岡 (0258)23−1151
- 長野 (0262)28−4618
- 上田 (0268)25−2788
- 松本 (0263)25−0547

シャープビジネス㈱宇都宮
〒320 宇都宮市不動前4丁目2番14号
☎(0286)37-3576

- 水戸 (0292)41−0552
- 土浦 (0298)24−2231
- 前橋 (0272)23−3511
- 太田 (0276)46−8255

東京地区

シャープビジネス㈱東京
〒130 東京都墨田区石原2丁目12番3号
☎(03)625-5111(代)

- 江東 (03) 625−5111
- 城東 (03) 620−9591
- 城西 (03) 384−7806
- 城南 (03) 737−1561
- 城北 (03) 958−3374
- 三多摩 (0425)84−1220
- 東京 (03) 436−0811
- 千葉 (0472)63−4043
- 松戸 (0473)68−0279
- 埼玉 (0486)63−5159
- 埼玉南 (0484)45−6671
- 熊谷 (0485)54−9011
- 横浜 (045)751−3215
- 平塚 (0463)54−2024
- 甲府 (0552)26−1035
- 静岡 (0542)82−8825
- 沼津 (0559)22−5673

北陸地区

シャープビジネス㈱金沢
〒921 石川県石川郡野々市町御経塚1096番1号
☎(0762)49-1240

- 富山 (0764)51−1317
- 福井 (0776)27−1800

中部地区

シャープビジネス㈱名古屋
〒454 名古屋市中川区山王3丁目5番5号
☎(052)332-2631

- 岐阜 (0582)73−2325
- 浜松 (0534)63−1551
- 豊橋 (0532)53−0648
- 津 (0592)32−4123
- 四日市 (0593)51−6703

近畿地区

シャープビジネス㈱大阪
〒556 大阪市浪速区恵美須町西1丁目2番9号
☎(06)643-3021

- 船場 (06) 271−0319
- 北大阪 (06) 328−0140
- 東大阪 (0729)94−1619
- 南大阪 (0722)41−9266
- 奈良 (07435)3−1423
- 和歌山 (0734)44−6130
- 京都 (075)661−7335
- 滋賀 (0775)43−2131
- 舞鶴 (0773)76−6037
- 神戸 (078)452−8531
- 姫路 (0792)66−8411

中国地区

シャープビジネス㈱広島
〒731-01 広島市安佐南区祇園町大字西原2249番地の1
☎(082)874-4925

- 東広島 (0824)28−3061
- 福山 (0849)51−3522
- 岡山 (0862)43−6644
- 松江 (0852)26−5211
- 鳥取 (0857)22−0621
- 徳山 (0834)31−5521
- 小郡 (08397)2−3881
- 下関 (0832)52−5508

四国地区

シャープビジネス㈱高松
〒760 高松市木太町1861番地の3
☎(0878)33-4255

- 松山 (0899)71−1146
- 新居浜 (0897)43−8840
- 徳島 (0886)25−2103
- 高知 (0888)83−7008

九州地区

シャープビジネス㈱福岡
〒816 福岡市博多区井相田2丁目12番地の1
☎(092)572-2611

- 北九州 (093)592−6521
- 佐賀 (0952)25−2640
- 長崎 (0958)46−9513
- 佐世保 (0956)32−6281
- 熊本 (0963)62−9036
- 大分 (0975)36−0289
- 宮崎 (0985)25−2251
- 延岡 (0982)31−2883
- 鹿児島 (0992)52−1659

沖縄地区

シャープビジネス㈱沖縄
〒900 沖縄県那覇市曙2丁目10番地の1
☎(0988)61-7360

シャープではソフトウェア情報小紙
「MZソフトウェアインフォメーション」を
随時発行しています。
マイコンショップに配布しています。
最寄りのお店でご利用ください。

気軽に楽しくコンピュータが学べる、良くわかると絶賛されたTV提供番組〝パソコンサンデー〟
ビデオカセット発売中!!
上巻・下巻:各15,000円

●お求めはお近くの有名書店、マイコンショップでどうぞ。

中部地区

朝日文化センター
☎(052)581-3631
●BASIC入門・上級コース

毎日文化センター
☎(052)581-1366
●マイコン入門・中級コース

名古屋電気通信工学院
☎(052)682-3189
●パソコン初級・中級コース

名古屋情報経理専門学校
☎(052)263-9500
●入門・初級・中級・上級コース

中部シャープマイコン教室
☎(052)332-2626
●マイコン入門コース

日経パソコン教室
☎(052)332-1831
●ビジネスコース

岐阜中日文化センター
☎(0582)65-2531
●マイコン初級・中級コース

近畿地区

KBS京都放送アカデミー
☎(075)441-4161
●BASIC入門A・Bコース

近鉄文化教室〈阿倍野〉
☎(06)649-0071
●BASIC入門A・Bコース
●ビジネスコース

朝日カルチャーセンター〈神戸〉
☎(078)321-5222
●BASIC入門コース

朝日カルチャーセンター〈大阪〉
☎(06)222-5222
●BASIC入門コース

アクセスイン
☎(06)644-6611
●BASIC入門コース

中国地区

中国新聞文化センター〈広島〉
☎(0822)47-4788
●BASIC入門A・Bコース

中国新聞文化センター〈福山〉
☎(0849)32-1362
●初級マイコン講座

山陰中央新報文化センター
☎(0852)26-3262
●実務マイコン入門コース

BSSファミリースクール〈山陰放送〉
☎(0859)33-0661
●マイコン教室

岩国家政専門学校
☎(0827)21-1566
●BASIC入門コース

四国地区

四国新聞社
☎(0878)33-1111
●BASIC入門・中級コース

南海放送〈松山〉
☎(0899)33-5151
●BASIC入門コース

九州地区

KBC朝日文化センター〈福岡〉
☎(092)713-1144
●BASIC入門コース

朝日文化センター〈北九州〉
☎(093)521-8381
●BASIC入門コース

朝日文化センター〈福岡〉
☎(092)431-7751
●BASIC入門コース

毎日文化センター〈北九州〉
☎(093)541-1181
●BASIC入門コース

宮崎ドレメマイコンスクール
☎(0985)25-9166
●BASIC入門・中級コース

SilverBall MZ-

Silver Ball（シルバーボウル）の「あそび」がきみの知
MZ-2000が楽しめるアメリカン・ソ

弾をさけての道路工事なんて。やってみるー？

MISSION Z2000
ミッション

A国より我U国を攻撃するため飛行隊が襲来した。君の使命は味方のエネルギー補給とその走路を死守することである。ターゲットスコープを使用し，敵を撃墜してほしい。必要な時は道路補修ができる。エネルギー補給がとだえると，GAME OVER！

SilverBall MZ-2000 COLOR USA版

スーパーマンになってみたかった。バキューン。

OMEGATRON
オメガトロン

君は敵に囲まれた部屋に閉じ込められてしまった。次々と現われる敵のロボットをGun, Laser, ORBの3種の武器を使いわけ，破壊してほしい。ORBは使うごとにエネルギーが減少するので注意！

戦争なんてイヤだ。でもゲームの中では面白い。

FLYBOY
フライボーイ

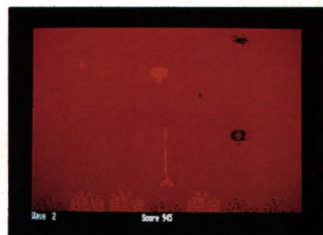
戦闘機のパイロットである君は，5機編隊で敵を攻撃するため出撃した。敵の攻撃をかわし，下を走るジープ，タンク，トラックを破壊してほしい。もし，撃墜された時は4機まで乗りつぐことができる。ただし，時々出てくる救急車を攻撃するとペナルティとなる。

2000 COLOR/USA版

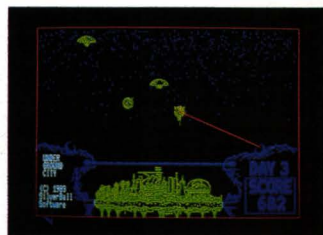
能に挑戦する。
フト新発売。

この町には「胸キュン」がいる。ガンバラなくっちゃ。

DAWN PATROL
ドーン パトロール

夜明け前の静けさを破って敵が我々の都市を攻撃してきた。敵は飛行機と気球で来襲し，爆弾を落とすので，君は砲台を操作し，飛行機・気球，爆弾をロケットで撃ち落としてほしい。すべての都市を破壊される前にどれだけ多く飛行機を撃ち落とせるか，健闘を祈る。

エイリアンの変化の面白さが、戦いを忘れさせた。

UNDERGROUND CITY
アンダーグラウンドシティ

エイリアンが地球征服を企み攻撃してきた。我々は都市を地下へ隠し防戦体制に入った。レーザービーム砲でエイリアンを攻撃し都市を守ってほしい。エイリアンはレーザービームが当たるごとに形を変化させる。エイリアンの攻撃が一発でも都市に当たると都市は地下深く沈んでしまう。エイリアンは毎日攻撃してくるが君は何日持ち堪えることができるだろうか。

本当に楽しいゲームは楽しさをしった者にしか出来ない。アメリカ生まれの面白ゲームが5本。どれでもよいからきみの脳細胞に反応したソフトをRUNしてみよう。
カラーのCRTを使うとゲームに一層迫力が出る。もちろん純正のCRTでも遊べる。
Silver Ballはこの5本からスタートして次々と新しいゲームソフトを発表していく。アメリカン・ソフトに挑戦してほしい。

定価　各巻共2,800円

このソフトはコンパイラでかかれています。

シルバーボールソフトウエア株式会社　バークレー，カルフォルニア，U.S.A.

キミの頭脳は高感度すぎて
時代はキミをクレイジーと呼ぶが、
私たちは、そのスーパークレイジーぶりに
期待している。
クレイジーと天才は紙一重、
1時代の差があるだけだ。
主役はいまスーパークレイジーの時代。

スーパークレイジーたちの祭典

PASOL SOFT CONTEST

ぱそる
ソフトコンテスト
賞金総額
400万円

ヴォルテージ

通信販売、店頭販売 8 大特典

特典1 沖縄の旅10名様ご招待

期間中1万円以上お買上げのお客様を抽選で沖縄の旅にご招待いたします。

（期間／58年5月20～9月30日）

お買上げ金額ごとに
豪華プレゼント

1. 30万円以上 ポケコンPC-1245
2. 20万円以上 ゲーム機器
3. 10万円以上 テープケース
4. 10万円未満5万円以上 生テープ5本

（期間／58年5月20日～9月30日）

現金・分割とも**超特価**奉仕

1万円以上お買上げのお客さまを、ソフトコンテスト受賞イベントへ抽選でご招待いたします。

（期間／58年5月20日～9月30日）

機能拡大、愉しみもまた拡大

●PC-8001mkII（本体） ~~123,000円~~
●FM-7（本体） ~~126,000円~~
●MZ-731（本体＋カセット・レコーダー＋カラー・プロッタ） ~~128,000円~~

（PC専用）

フロッピーディスク（PC-80S31, NEC）	~~168,000円~~
シンセサイザー（CMU-800, アムデック）	~~65,000円~~
シンセサイザー（PCS-8007, パックエレクトロ）	~~24,800円~~
音声発生（Mr.ボイス, テックメイト）	~~49,000円~~
音声発生（読み上げさん, 和幸舎）	~~35,000円~~

（PC, FM兼用）

プリンター（GP-250F, 精工舎）	~~59,800円~~
プリンター（GP-700M〔カラー〕, 精工舎）	新製品
プリンター（GP-550E〔漢字〕, 精工舎）	新製品
プリンター（RP-80, EPSON）	~~89,000円~~
プリンター（FP-80, EPSON）	~~149,800円~~

プロッター（マイプッロットJr, 渡辺測器）	~~99,800円~~
フロッピーディスク（TF-20, EPSON）	~~142,000円~~
カラーCRT（FTC-1416, 東映通商）	~~63,000円~~
カラーCRT（FTC-1425〔高解〕, 東映通商）	~~105,000円~~
デジタイザー（K-510, 関東電子）	~~148,000円~~

募集要項

応募作品

①ゲームソフトおよび教育用ソフト
（教育用ソフトはゲーム要素の強いもの）
②ゲームおよび教育用ソフトのアイディア
両部門とも、自作の未発表作品に限ります。

応募条件

使用機種は、メーカーは問いませんが、特殊なものは除きます。使用言語はBASICまたはマシン語とし、プログラムはカセットテープ、ミニディスケットのどちらでもかまいません。

応募方法

①作品には、使用機種、簡単なプログラムの紹介(名称等)と操作方法ならびにプログラムリスト、サンプルデータ等を添付のこと。
②アイディア作品については、3画面以上のカラーイラストとシナリオを添付すること。
③応募者は作品に、氏名・住所・電話番号・年令・職業（学生の場合は学校名）を明記。また簡単な自己紹介を添付のこと。

応募締切

昭和58年9月10日（当日消印有効）
発表
入選者は10月末に直接本人に通知するほか、店頭、本誌12月号および各専門誌上にて発表します。

賞

総額400万円
●ソフト部門
最優秀賞1点　賞金　150万円
優 秀 賞2点　賞金　各50万円
佳　　作5点　賞金　各15万円

●アイディア部門
最優秀賞　1点　賞金　30万円
優 秀 賞　3点　賞金　各7万円
佳　　作12点　賞金　各2万円

審査

当社開発部、各メーカー担当者およびゲスト審査員により、厳正な審査を行ない、各賞を決定します。
●ゲスト審査員　所ジョージ・石川ひとみ

作品の送り先と問合せ先
〒107 東京都港区赤坂1-5-11　新虎ノ門ビル
㈱ぱそる開発部「ソフトコンテスト」係
☎03(588)-1717　担当　辻本
※入賞作品の版権はすべて当社に帰属します。
また応募作品は原則としてお返しいたしませんのでご了承ください。返却希望者は、郵送料を添えてご応募ください。

…は12月にやってくる。

受賞式は華々しくにぎやかに開催します。お祝いに駆けつけるゲストの顔ぶれもすごい。楽しみは12月までおあづけだけど、待てるかな。

●ゲスト 石川ひとみ

●ゲスト審査員 所ジョージ

●司会　木藤隆雄
（ドレミファドン レギュラー）

日時　昭和58年12月10日(土)
P.M1:00～4:00
会場　「ぱそる」店内

受賞式の内容が一部変更になる場合もあります。あらかじめご了承ください。

特典5 配送費無料

特典6 修理費(配送費とも)無料

特典7 日本信販との提携ローンで最長36回まで分割可能

特典8 テレフォンサービス実施
技術指導、ご質問等お気軽にお電話ください。
TEL 03(588)1717
（平日a.m10：00～p.m7：00
土曜a.m10：00～p.m5：00
＊日・祭日はお休みです）

全国の皆さまへぱそる通信販売

電話またはハガキにてお申し込みください。
どこへでも即納（配送無料）します。ハガキでお申し込みの場合 ①機種名 ②お支払い方法（現金またはクレジット回数）③氏名・住所・電話番号・年令・職業をご明記ください。

ゆったり、たっぷり
260m²の体験空間。

(FM専用)
フロッピーディスク(薄型フロッピー, 富士通)　107,000円
プリンター(MB27404, 富士通)　89,000円
(MZ専用)
カラーCRT(MZ-1D05, シャープ)　69,800円
ジョイスティック(Joy-700, 九十九電気)　5,800円

本社・ショップ 〒107東京都港区赤坂1-5-11
(新虎ノ門ビル1F)　TEL03(588)1717(代)
和歌山営業所 〒624 和歌山県海南市藤白758番地
富士興産(株)内　TEL07348(2)7332(代)

カタログご希望の方は、ハガキに氏名・住所・年令・職業をご明記のうえ、カタログ請求券を貼ってお申し込みください。

カタログ請求券 oh! MZ 7月号

Hudson soft

遂に完成 HuWP
XI用日本語ワープロ堂々完成
定価 ¥24,800

新作ゲームソ

ゲームソフト

4人マージャン

1 4人マージャン 誌上初公開!!

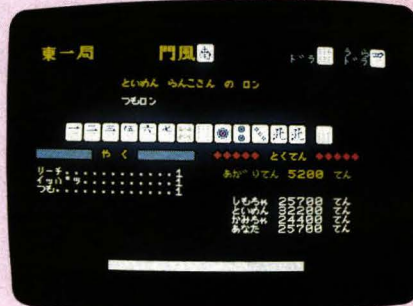
2 さあいよいよ リーチ

ついに本格的4人麻雀ゲームが登場しました。下家マコト君、対面ミキちゃん、上家タケシ君というおなじみのメンバーと、東々まわしの半チャンでもやりましょうか。ただし二ハンしばり流し満貫とカンはなしのハドソン・ルールです。もちろん3人の捨牌を見ることもOK！マージャン技術のレベルアップにどうぞ。

X-1047-G 言語M ¥4,800

3 すてハイに気をつけて ふりテンに注意しよう。

5 あまりあせると チョンボもありますよ。

4 点数も自動的に 出てきます。

ザ・スパイダー
X-1050-G 言語M ¥3,800

華麗な怪盗
X-1055-G 言語M ¥3,800

MJ-05
X-1053-G 言語M ¥3,800

ひつじや〜い
X-1052-G 言語M ¥3,800

JUSTICE KNIGHT
X-1056-G 言語M ¥3,800

HELP!
X-1054-G 言語M ¥3,800

※最新ソフトカタログをご希望の方はX1,MZ-700,FM-7,MZ-2000,PC-8001mkIIの機種を明記の上ハドソン東京まで〒60円を添えて、お申し込み下さい。
※通信販売はソフト名、SERIAL No.を明記の上お近くのハドソンZ-7係までお送り下さい。
※NO表示のGはグラフィックRAMが必要です。

001mkII用特選☆富士通FM-7用特選

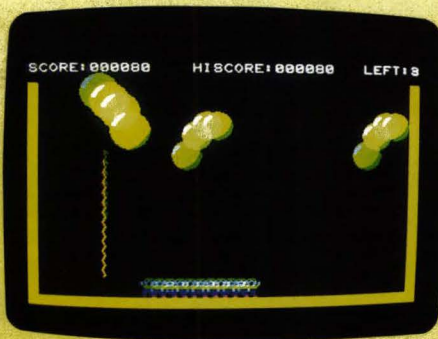
フト誌上公開

miniHuBASICコンパイラー好評発売中
（整数型ベーシック→マシン語変換プログラム）
適応機種，XI, MZ-700, PC-8001mkII
定価　¥6,000

ゲームソフト

キャノンボール

1 密室の中にボールが1・2・3……

2 ボールがふえてきました。高得点の秘密はボールを一つずつ確実に消すことです。

3 ほとんどボールの洪水、貴方の運命は……

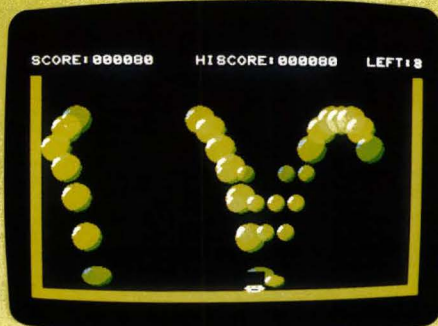
君はある時ふと目をさますと狭い部屋に閉じ込められていた。すると突然頭上から大きなボールが落ちてきた。君のもっている武器は槍だけだ。ボールにつぶされないようにボールを破壊しなければならない。ボールは破壊されるたびに2つに分かれて、しまいに消えてしまうが下手に破壊すると部屋中ボールだらけになってしまう。そこは君の頭脳でカバーしてくれ！

XI	X-1033-G	言語M	¥3,800
PC8001mkII	YB-1008	言語M	¥3,800
FM-7	LA-1030	言語M	¥3,800

爆弾男

1 さあゲームスタート

2 フーセンオバケをおびきだし爆弾をしかけよう。

3 タイミングよく、爆弾をしかけフーセンオバケをやっつけよう。

フワフワ、ポヨ〜ンと無気味な風船オバケにつかまったら最後、あなたは食べられてしまう。しかし!!あなたは爆弾を沢山持っている。タイミングよくセットして、爆風で風船オバケを割ってしまうのだ。そして次々に壁をこわして宝物を見つけよう!!ただし、壁の中には風船オバケも眠っている。中から出て来るのは、どっちだ!?

XI	X-1031-G	言語M	¥3,800
PC8001mkII	YB-1003	言語M	¥3,200
FM-7	LA-1023	言語M	¥3,200

あらゆるメディアに人間性を
HUDSON GROUP
HUDSON SOFT
ハドソン札幌/〒062 札幌市豊平区平岸3条5丁目4番17号 コロナード平岸II 201 PHONE:011-821-1538
ハドソン東京/〒102 東京都千代田区麹町4丁目7番5号 麹町ロイヤルビル2F PHONE:03-234-4996
ハドソン大阪/〒542 大阪市南区南船場4丁目2番18号 佐野屋橋ビル5F PHONE:06-251-1945
ハドソンUSA/2063 CENTER STREET BERKELEY CA 94704 TELEPHONE 415-845-1416

あなたに大接近。

ひき続き 加盟店募集中！

特別加盟店 現在市販されているソフトのほとんどを揃えています。

■ **第一家電OA販売株式会社**
〒101 千代田区神田佐久間町1-15
03-253-0808〈担当・飯田〉

■ **西武百貨店池袋店マイコンショップ**
〒171 豊島区南池袋1-28-1
03-981-0111 内線2928〈担当・池田、鈴木〉

■ **ラオックス新宿店マイコンシティ**
〒160 新宿区新宿3-15-16 ラオックスビル5F
03-350-1241〈担当・菅原店長〉

■ **ヤマギワテクニカ**
〒101 千代田区外神田4-3-1
03-253-0121(代)〈担当・原〉

■ **大塚OAセンター八重洲**
〒104 中央区八重洲2-1-5 東京駅前ビル2
03-281-4711〈担当・金子〉

■ **J&Pテクノランド**
〒556 大阪市浪速区日本橋5-6-7
06-644-1413〈担当・勝田店長〉

■ **J&Pメディアランド**
〒556 大阪市浪速区日本橋4-11-3
06-644-1613〈担当・山本店長〉

■ **マイコンショップCSK**
〒530 大阪市北区梅田1-1-3 大阪駅前第3ビルB-1
06-345-3351〈担当・四元〉

■ **YDKシステムセンター**
〒160 新宿区西新宿2-4-1 新宿NSビル1F
03-342-2441〈担当・宮本店長〉

■ **マイコンショップCSK(西口)**
〒160 新宿区西新宿1-12-18(勧業角丸証券隣)
03-342-1901〈担当・前田店長〉

ソフトバンク加盟店なら、あなたの欲しいソフトのほとんどが入手できます。お店の方にご相談ください。

■ **J&P渋谷店**
〒150 渋谷区道玄坂2-28-4 井門ビル
03-496-4141

■ **J&P京都店**
〒600 京都市下京区寺町通仏光寺下ル恵美須之町549
075-341-3571

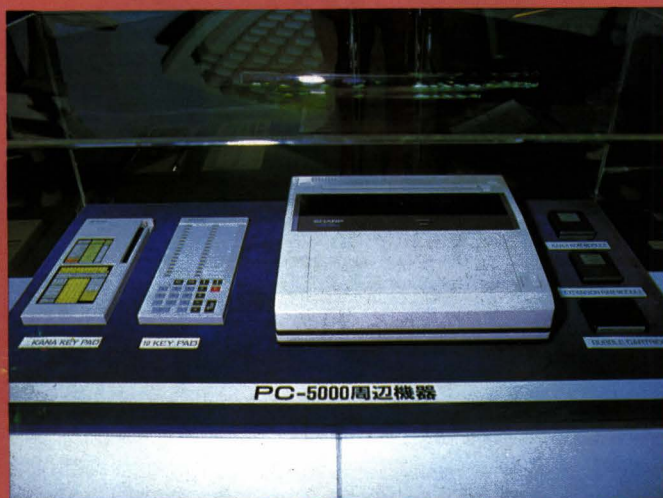
ソフトバンクにご連絡下さい。あなたのお近くのソフトバンク加盟店をご紹介します。

株式会社 **日本ソフトバンク** SOFT BANK
〒102 東京都千代田区九段南2-3-11 靖国九段南ビル 03(263)3690
〒542 大阪市南区難波千日前5-19 河原センタービル 06(644)0191

The 58th International Business Show & Micro Computer Show '83

世はまさにコンピュータ時代というわけで、ますます熱のこもった、初夏のコンピュータの２大ショウビジネスショウ（５月11日〜14日）とマイコンショウ（５月25日〜28日）が開催されました。あちこちのメーカーが、このショウに向けて新製品を発表するとあって、コンピュータ界の動向をいち早く知ろうという観衆でにぎわいました。ここで気になるのがシャープの新製品、超目玉は「MZ-2200」。自由に組めるコンポタイプとの宣伝文句どおり、データレコーダ、ディスプレイが分離し、今までにないMZの顔を見せてくれました。合わせて、人工知能言語LOGOも登場し、MZの今後が注目されるといったところでしょう（新製品ニュースは、本文128ページ〜にあります）。

❶ブース正面。通勤ラッシュ並みの人だかりでした。

❷新製品PC-5000の紹介に足を止める。

❸PC-5000と周辺機器たち。

❹PC-5000のデモ。液晶ディスプレイとプリンタが付いて、ちょっと大きな電卓って感じでしょ。

❺❻MZ-2000とMZ-700もパーソナルなビジネスアシスタントとしてお使いください。

❻

❽まずはポケコンからというわけで黒山の人だかりです。

❼MZ-2000とPC-1500をつないで……。

❾❿ビジネスショウですから、シャープのOA機器。

❿

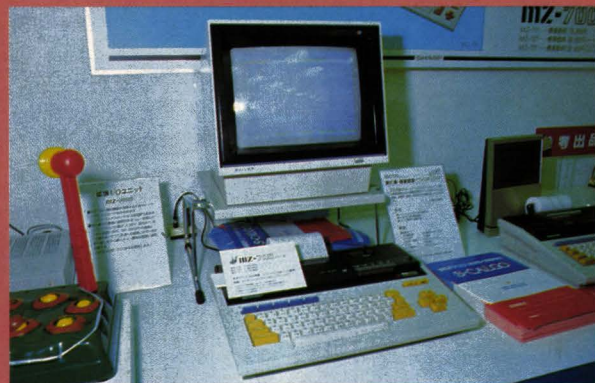
⓫第１会場の入口入って真正面、新製品MZ-2200、PC-5000、16ビットボードキットの紹介コーナーは大盛況。

⓬MZシリーズの新顔、MZ-2200を紹介する美女２人。

⓭MZ-2200システムの図。５インチのミニフロッピードライブ（MZ-１F07）と80桁ドットプリンタ（MZ-１P07）。

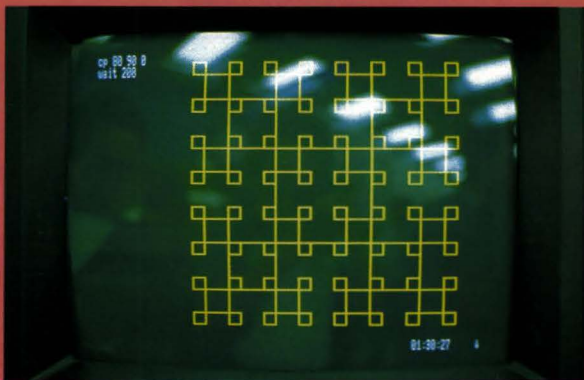
⓮３インチのコンパクトフロッピーディスク（MZ-１F06）。価格は未定です。

⓯MZ-700、液晶ディスプレイでゲームに熱中してました。

⓰ビジネスショウで初お目見えのPC-5000。

⓱東海クリエイトの簡易言語「S・CALGO」をデモってました。

⑱パソコンテレビX I のコーナー。これは遊べる。

⑲ポケコンのコーナーです。

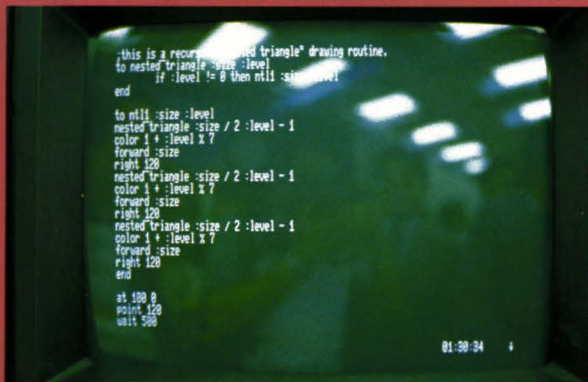
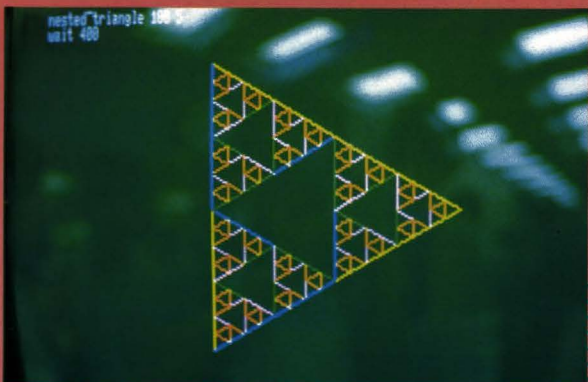
⑳やった〜！MZ。MZ-LOGO新登場の解説パネルです。

㉑㉒㉓LOGOサンプルプログラムのデモ

㉒

㉔「パソコンサンデー」でお馴じみの宮永先生も登場です。

㉓

マイクロコンピュータ技術者募集

15

WELCOME **SUPER STAFF**

日本ソフトバンクでは、出版事業等の拡張により、マイクロコンピュータ技術者および英文技術翻訳者を以下の要領で大募集いたします。秘密は厳守しますので、どしどしご応募ください。

■職　種■

①マイコン雑誌の企画担当技術者(10名)

●アセンブラでプログラムを作れる方●インタープリタ／コンパイラを理解できる方●ハードの回路図を理解できる方●原稿の書ける方

②海外事業関係(5名)

●技術英文の翻訳ができ、マイコンに興味のある方

■応募方法■

履歴書(写真貼付)を7月末日までに下記あてに郵送して下さい。追って面接日を通知します。なお、①についてはアセンブラおよびインタープリタ／コンパイラによる自作プログラムを同封してください。応募の秘密は厳守します。

■待　遇■

当社規定による。①については、マイコン雑誌経験者を特に優遇します。

■あて先■

〒102 東京都千代田区九段南2-3-14 靖国九段南ビル2F
日本ソフトバンク技術室

正社員でもアルバイトでも可。アルバイトの場合は自宅勤務も可能です。

SOFT BANK

株式会社 **日本ソフトバンク**
〒102 東京都千代田区九段南2-3-14 03(263)3664
〒542 大阪市南区難波千日前5-19

●表紙のマシン MZ-2200

Oh!MZ 1983 JUL.

表紙原画／マジック・バス
表紙写真／浜崎　昭

特集　マシン語プログラミング

BASIC内サブルーチンとしてのマシン語プログラム

長瀬 敏之

まえがき

BASIC言語でプログラムを作成するとき，次のような問題を生ずることがあります。

①ある処理を行ないたいけれども，BASIC言語の中には，それを行なう命令がない。

②ある処理を行なうには，時間がかかりすぎて，実用にならない。

このような問題は，マシン語を使用すればたいてい解決できます。

そこで，このような問題が発生した場合の今後の参考に，ということで，ここでは例として，マシン語サブルーチンを２種類紹介します。

A. マシン語サブルーチン

プログラム　1

●TV画面の座標（X１，Y１）－(X２，Y２）を対角とする長方形内を，アスキーコード（Z１）のキャラクタで埋めるマシン語サブルーチンです。サブルーチンの実行前に，必要なデータ（X１，Y１，X２，Y２，Z１）を，B040H番地以降に書き込んでおいてください。

●各データの値の範囲は，次のとおりです。

0≦(X１)≦(X２)≦39or79

0≦(Y１)≦(Y２)≦24

0≦(Z１)≦255

●注：MZ-80K/C，MZ-1200，MZ-700については，(Z１)は，アスキーコードではなく，ディスプレイコードです。

●**PROGRAM　1－A**

MZ-80K/C，MZ-1200用

MZ-700(80K/Cモード)用

●**PROGRAM　1－B**

MZ-700用

●**PROGRAM　1－C**

MZ-80B用

●**PROGRAM　1－D**

MZ-2000用

●各プログラムとも，実行開始番地はB000H番地です。

プログラム　2

このマシン語サブルーチンは,「ソート」プログラムの一種で，N個のデータを昇順(値の小さい順）に並べ替えます。

このマシン語サブルーチンを実行する前に，あらかじめ，データの数(N)をB070～B071H番地に，N個のソートすべきデータをB072H番地以降に書き込んでおいてください。

サブルーチン実行後には，N個のデータがB072H番地以降に昇順にソートされています。

●データの値の範囲は,「メモリの２バイト分」(0000H～FFFFH，0～65535)です。Nの値の範囲は,「B072H番地以降のフリーエリア／2」以下です。

例　MZ-1200の場合のNの値

$$N \leqq 4039\left(=\frac{\text{CFFFH}-\text{B072H}-1}{2}\right)$$

●データの値は２バイトですから，たとえば，データの数(N)が３個で，データが0123H，FEDCH，7777Hならば，メモリには次のように書き込みます。

(B070H)＝03H　(B074H)＝DCH

(B071H)＝00H　(B075H)＝FEH

(B072H)＝23H　(B076H)＝77H

(B073H)＝01H　(B077H)＝77H

●**PROGRAM　2**

CPUが「Z-80」で，B000H番地以降にフリーエリアがあるコンピュータ用

●プログラムの実行開始番地はB000H番地です。

B. プログラム使用例

実際に前述のプログラムを使用する人のために，実例を挙げておきますから，参考にしてください。実例で使用しているBASICは，Hu-BASICです。

注：「＄00」，「00H」，「＆H00」の「＄」，「H」，「＆H」はどれも16進数を示す記号です。

どれを使用するかは，BASICの種類によって異なります。

PROGRAM　1（BASIC）

このプログラムを実行すると，座標(10，10)－(10，20)－(20，20)－(20，10)の長方形の中が，アスキーコード80(50H)のキャラクタで埋まります。

もちろん,「PROGRAM　1」(マシン語)がメモリ内にすでに書かれているとします。このとき,「LIMIT　＆HAFFF」または「CLEAR　＆HAFFF」を実行して，マシン語エリアを確保しておいてください。これは，マシン語サブルーチンがBASICによって壊されないようにするためです。「PROGRAM　2」についても，同様なことを行なってください。

PROGRAM　2（BASIC」

乱数を使って，100個のデータを作り，それをソートします。

文番号　210　の「GOSUB　1000」を，「GOSUB　2000」に書き直せば，マシン語サブルーチンを使用しないで，BASICプログラムのみでソートを行ないます。これにより，ソートの所要時間を測定したところ次のような結果になりました。

データ数	BASIC	マシン語
N＝ 100	00：00：49	00：00：06
N＝ 500	00：19：55	00：00：40
N＝1000	01：19：39	00：01：45

・XX:XX:XXは，時：分：秒を表す。
・使用機種　MZ-80K
・クロック　2MHz
・マシン語サブルーチンの実行時間にはPOKE，PEEKの時間も含まれていますから，実際にソートを行なっている時間は，もっと短くなります。
・クロックが4MHzのマイコンなら，(たとえば，MZ-2000)所要時間は，半分ぐらいになります。

C. プログラムの解説

●PROGRAM　1－A

10行目～　長方形の縦横の長さを求める。
19行目～　(X1，Y1)のV-RAM上での番地を求める。
31行目～　長方形の内側に相当するV-RAMエリアに，(Z1)の値を書き込む。

●PROGRAM　1－B

10行目と50行目は，それぞれV-RAMをメインメモリにつないだり切り離したりする命令です。この命令の詳細は，「MZ-700編」として連載の方で説明する予定ですから，しばらく（4～5ヶ月）お待ちください。他の部分は，「PROGRAM　1－A」と同じです。

●PROGRAM　1－C

V-RAMをメインメモリにつないだり，切り離したりする命令（連載にて説明済みです）が，プログラムの前後に追加してある他は，「PROGRAM　1－A」と同じです。このプログラムは，40キャラクタモード用ですが，28・29行目[　]内の値を使用すれば，80キャラクタモード用になります。具体的には，次のようにします。

・LD　HL，CFD8H
　↓
　LD　HL，CFB0H
　LD　DE，0028H
　↓
　LD　DE，0050H

●PROGRAM　1－D

「PROGRAM　1－C」と，ほぼ同じです。

●PROGRAM　2

このプログラムは，「PROGRAM　2」(BASIC) の文番号2000～2180のBASICプログラムに，無理やり(?)似せて作りましたから，それと比較してもらうと，このプログラムがよくわかります。BASICの文番号「2100」，「2110」，……は，アセンブラのラベル「LP100」，「LP110」，…に相当し，変数「JA」，「JB」，「XA」は，レジスタ「BC」，「IY」，「IX」に相当し，変数「A(0)」，「A(1)」，…は，メモリ「B070～1H」，「B072～3H」，…に相当します。

なお，ここで使用したソート・プログラムは，「直接選択法」と呼ばれるものです。

●PROGRAM　1 (BASIC)

特に説明することはないと思います。

●PROGRAM　2 (BASIC)

100 ～　変数Aにデータを代入する
150 ～　ソート前のデータ表示
200 ～　ソート・プログラムを呼ぶ
300 ～　ソート後のデータの表示
1100～　メモリに，データの数とソート前のデータを書き込む
1200～　マシン語サブルーチンを呼ぶ
1300～　メモリから，ソート後のデータを読み出す
2000～　BASIC言語によるソート・プログラム

・データの数を変更するときは，最初の部分
　100　　DIM　A(100)
　110　　A(0)＝100
の，数値「100」を変更してください。

```
      **  Z80 ASSEMBLER SP-2102 PAGE 01  **

    01 0000                  ;
    02 0000                  ;  PROGRAM 1-A
    03 0000                  ;
    04 0000                  ;     V-RAM CLR (X1,Y1)-(X2,Y2)←(Z1)
    05 0000                  ;
    06 0000                  ;        MZ-80K/C,MZ-1200
    07 0000                  ;
    08 0000                     REL   B000H           ; ORG B000H
    09 B000                  ;
    10 B000 2A40B0              LD    HL,(X1)         ; H=Y1,L=X1
    11 B003 EB                  EX    DE,HL           ; SWAP DE,HL
    12 B004 2A42B0              LD    HL,(X2)         ; H=Y2,L=X2
    13 B007 A7                  AND   A               ; CY=0
    14 B008 ED52                SBC   HL,DE           ; HL=HL-DE-CY=HL-DE
    15 B00A 24                  INC   H               ; H=H+1
    16 B00B 2C                  INC   L               ; L=L+1
    17 B00C E5                  PUSH  HL              ; (SP)=HL
    18 B00D                  ;
    19 B00D 2A40B0              LD    HL,(X1)         ; H=Y1,L=X1
    20 B010 44                  LD    B,H             ; B=H
    21 B011 4D                  LD    C,L             ; C=L
    22 B012 04                  INC   B               ; B=B+1
    23 B013 21D8CF              LD    HL,CFD8H        ; HL=CFD8H
    24 B016 112800              LD    DE,0028H        ; DE=0028H
    25 B019               LP01:  ENT
    26 B019 19                  ADD   HL,DE           ; HL=HL+DE
    27 B01A 10FD                DJNZ  LP01            ; B=B-1
    28 B01C                  ;                          IF B<>0 GOTO LP01
    29 B01C 09                  ADD   HL,BC           ; HL=HL+BC
    30 B01D                  ;
    31 B01D C1                  POP   BC              ; BC=(SP)
    32 B01E 3A44B0              LD    A,(Z1)          ; A=(Z1)
    33 B021               LP02:  ENT
    34 B021 C5                  PUSH  BC              ; (SP)=BC
    35 B022 E5                  PUSH  HL              ; (SP)=HL
    36 B023 41                  LD    B,C             ; B=C
    37 B024               LP03:  ENT
    38 B024 77                  LD    (HL),A          ; (HL)=A
    39 B025 23                  INC   HL              ; HL=HL+1
    40 B026 10FC                DJNZ  LP03            ; B=B-1
    41 B028                                             IF B<>0 GOTO LP03
    42 B028 E1                  POP   HL              ; HL=(SP)
    43 B029 19                  ADD   HL,DE           ; HL=HL+DE
    44 B02A C1                  POP   BC              ; BC=(SP)
    45 B02B 10F4                DJNZ  LP02            ; B=B-1
```

```
46 B02D                   ;                                    IF B<>0 GOTO LP02
47 B02D C9                          RET                      ; RETURN
48 B02E                   ;
49 B02E                             DEFS  12H
50 B040                   ;
```

```
    **  Z80 ASSEMBLER SP-2102 PAGE 02  **

01 B040                   X1:       ENT
02 B040 00                          DEFB  00H
03 B041                   Y1:       ENT
04 B041 00                          DEFB  00H
05 B042                   X2:       ENT
06 B042 00                          DEFB  00H
07 B043                   Y2:       ENT
08 B043 00                          DEFB  00H
09 B044                   Z1:       ENT
10 B044 00                          DEFB  00H
11 B045                   ;
12 B045                             END
```

```
    **  Z80 ASSEMBLER SP-2102 PAGE 01  **

01 0000                   ;
02 0000                   ;   PROGRAM 1-B
03 0000                   ;
04 0000                   ;       V-RAM CLR (X1,Y1)-(X2,Y2)+(Z1)
05 0000                   ;
06 0000                   ;          MZ-700
07 0000                   ;
08 0000                             REL   B000H                  ; ORG B000H
09 B000                   ;
10 B000 D3E3                        OUT   (E3H),A                ; V-RAM (ADR)
11 B002                   ;
12 B002 2A40B0                      LD    HL,(X1)                ; H=Y1,L=X1
13 B005 EB                          EX    DE,HL                  ; SWAP DE,HL
14 B006 2A42B0                      LD    HL,(X2)                ; H=Y2,L=X2
15 B009 A7                          AND   A                      ; CY=0
16 B00A ED52                        SBC   HL,DE                  ; HL=HL-DE-CY=HL-DE
17 B00C 24                          INC   H                      ; H=H+1
18 B00D 2C                          INC   L                      ; L=L+1
19 B00E E5                          PUSH  HL                     ; (SP)=HL
20 B00F                   ;
21 B00F 2A40B0                      LD    HL,(X1)                ; H=Y1,L=X1
22 B012 44                          LD    B,H                    ; B=H
23 B013 4D                          LD    C,L                    ; C=L
24 B014 04                          INC   B                      ; B=B+1
25 B015 21D8CF                      LD    HL,CFD8H               ; HL=CFD8H
26 B018 112800                      LD    DE,0028H               ; DE=0028H
27 B01B                   LP01:     ENT
28 B01B 19                          ADD   HL,DE                  ; HL=HL+DE
29 B01C 10FD                        DJNZ  LP01                   ; B=B-1
30 B01E                   ;                                        IF B<>0 GOTO LP01
31 B01E 09                          ADD   HL,BC                  ; HL=HL+BC
32 B01F                   ;
33 B01F C1                          POP   BC                     ; BC=(SP)
34 B020 3A44B0                      LD    A,(Z1)                 ; A=(Z1)
35 B023                   LP02:     ENT
36 B023 C5                          PUSH  BC                     ; (SP)=BC
37 B024 E5                          PUSH  HL                     ; (SP)=HL
38 B025 41                          LD    B,C                    ; B=C
39 B026                   LP03:     ENT
40 B026 77                          LD    (HL),A                 ; (HL)=A
41 B027 23                          INC   HL                     ; HL=HL+1
42 B028 10FC                        DJNZ  LP03                   ; B=B-1
43 B02A                   ;                                        IF B<>0 GOTO LP03
44 B02A E1                          POP   HL                     ; HL=(SP)
45 B02B 19                          ADD   HL,DE                  ; HL=HL+DE
46 B02C C1                          POP   BC                     ; BC=(SP)
47 B02D 10F4                        DJNZ  LP02                   ; B=B-1
48 B02F                   ;                                        IF B<>0 GOTO LP02
49 B02F                   ;
50 B02F D3E1                        OUT   (E1H),A                ; V-RAM (ADR OFF)
```

```
    **  Z80 ASSEMBLER SP-2102 PAGE 02  **

01 B031                   ;
02 B031 C9                          RET                          ; RETURN
03 B032                   ;
04 B032                             DEFS  0EH
05 B040                   ;
06 B040                   X1:       ENT
07 B040 00                          DEFB  00H
08 B041                   Y1:       ENT
09 B041 00                          DEFB  00H
10 B042                   X2:       ENT
11 B042 00                          DEFB  00H
12 B043                   Y2:       ENT
13 B043 00                          DEFB  00H
14 B044                   Z1:       ENT
15 B044 00                          DEFB  00H
16 B045                   ;
17 B045                             END
```

```
    **  Z80 ASSEMBLER SP-2102 PAGE 01  **

01 0000                 ;
02 0000                 ;  PROGRAM 1-C
03 0000                 ;
04 0000                 ;     V-RAM CLR (X1,Y1)-(X2,Y2)←(Z1)
05 0000                 ;
06 0000                 ;        MZ-80B
07 0000                 ;
08 0000                         REL   B000H            ; ORG B000H
09 B000                 ;
10 B000 DBE8                    IN    A,(E8H)
11 B002 CBFF                    SET   7,A              ; V-RAM (ADR)
12 B004 CBB7                    RES   6,A              ; V-RAM (D000H-)
13 B006 D3E8                    OUT   (E8H),A
14 B008                 ;
15 B008 2A40B0                  LD    HL,(X1)          ; H=Y1,L=X1
16 B00B EB                      EX    DE,HL            ; SWAP DE,HL
17 B00C 2A42B0                  LD    HL,(X2)          ; H=Y2,L=X2
18 B00F A7                      AND   A                ; CY=0
19 B010 ED52                    SBC   HL,DE            ; HL=HL-DE-CY=HL-DE
20 B012 24                      INC   H                ; H=H+1
21 B013 2C                      INC   L                ; L=L+1
22 B014 E5                      PUSH  HL               ; (SP)=HL
23 B015                 ;
24 B015 2A40B0                  LD    HL,(X1)          ; H=Y1,L=X1
25 B018 44                      LD    B,H              ; B=H
26 B019 4D                      LD    C,L              ; C=L
27 B01A 04                      INC   B                ; B=B+1
28 B01B 21D8CF                  LD    HL,CFD8H         ; HL=CFD8H [CFB0H]
29 B01E 112800                  LD    DE,0028H         ; DE=0028H [0050H]
30 B021                 LP01:   ENT
31 B021 19                      ADD   HL,DE            ; HL=HL+DE
32 B022 10FD                    DJNZ  LP01             ; B=B-1
33 B024                 ;                                IF B<>0 GOTO LP01
34 B024 09                      ADD   HL,BC            ; HL=HL+BC
35 B025                 ;
36 B025 C1                      POP   BC               ; BC=(SP)
37 B026 3A44B0                  LD    A,(Z1)           ; A=(Z1)
38 B029                 LP02:   ENT
39 B029 C5                      PUSH  BC               ; (SP)=BC
40 B02A E5                      PUSH  HL               ; (SP)=HL
41 B02B 41                      LD    B,C              ; B=C
42 B02C                 LP03:   ENT
43 B02C 77                      LD    (HL),A           ; (HL)=A
44 B02D 23                      INC   HL               ; HL=HL+1
45 B02E 10FC                    DJNZ  LP03             ; B=B-1
46 B030                 ;                                IF B<>0 GOTO LP03
47 B030 E1                      POP   HL               ; HL=(SP)
48 B031 19                      ADD   HL,DE            ; HL=HL+DE
49 B032 C1                      POP   BC               ; BC=(SP)
50 B033 10F4                    DJNZ  LP02             ; B=B-1
```

```
    **  Z80 ASSEMBLER SP-2102 PAGE 02  **

01 B035                 ;                                IF B<>0 GOTO LP02
02 B035                 ;
03 B035 DBE8                    IN    A,(E8H)
04 B037 CBBF                    RES   7,A              ; V-RAM (ADR OFF)
05 B039 D3E8                    OUT   (E8H),A
06 B03B                 ;
07 B03B C9                      RET                    ; RETURN
08 B03C                 ;
09 B03C                         DEFS  04H
10 B040                 ;
11 B040                 X1:     ENT
12 B040 00                      DEFB  00H
13 B041                 Y1:     ENT
14 B041 00                      DEFB  00H
15 B042                 X2:     ENT
16 B042 00                      DEFB  00H
17 B043                 Y2:     ENT
18 B043 00                      DEFB  00H
19 B044                 Z1:     ENT
20 B044 00                      DEFB  00H
21 B045                 ;
22 B045                         END
```

```
    **  Z80 ASSEMBLER SP-2102 PAGE 01  **

01 0000                 ;
02 0000                 ;  PROGRAM 1-D
03 0000                 ;
04 0000                 ;     V-RAM CLR (X1,Y1)-(X2,Y2)←(Z1)
05 0000                 ;
06 0000                 ;        MZ-2000
07 0000                 ;
08 0000                         REL   B000H            ; ORG B000H
09 B000                 ;
10 B000 DBE8                    IN    A,(E8H)
11 B002 CBFF                    SET   7,A              ; V-RAM (ADR)
12 B004 CBF7                    SET   6,A              ; V-RAM (ｷｬﾗｸﾀ)
```

```
13 B006 D3E8                    OUT    (E8H),A
14 B008                 ;
15 B008 2A40B0                  LD     HL,(X1)          ; H=Y1,L=X1
16 B00B EB                      EX     DE,HL            ; SWAP DE,HL
17 B00C 2A42B0                  LD     HL,(X2)          ; H=Y2,L=X2
18 B00F A7                      AND    A                ; CY=0
19 B010 ED52                    SBC    HL,DE            ; HL=HL-DE-CY=HL-DE
20 B012 24                      INC    H                ; H=H+1
21 B013 2C                      INC    L                ; L=L+1
22 B014 E5                      PUSH   HL               ; (SP)=HL
23 B015                 ;
24 B015 2A40B0                  LD     HL,(X1)          ; H=Y1,L=X1
25 B018 44                      LD     B,H              ; B=H
26 B019 4D                      LD     C,L              ; C=L
27 B01A 04                      INC    B                ; B=B+1
28 B01B 21D8CF                  LD     HL,CFD8H         ; HL=CFD8H [CFB0H]
29 B01E 112800                  LD     DE,0028H         ; DE=0028H [0050H]
30 B021                 LP01:   ENT
31 B021 19                      ADD    HL,DE            ; HL=HL+DE
32 B022 10FD                    DJNZ   LP01             ; B=B-1
33 B024                 ;                                 IF B<>0 GOTO LP01
34 B024 09                      ADD    HL,BC            ; HL=HL+BC
35 B025                 ;
36 B025 C1                      POP    BC               ; BC=(SP)
37 B026 3A44B0                  LD     A,(Z1)           ; A=(Z1)
38 B029                 LP02:   ENT
39 B029 C5                      PUSH   BC               ; (SP)=BC
40 B02A E5                      PUSH   HL               ; (SP)=HL
41 B02B 41                      LD     B,C              ; B=C
42 B02C                 LP03:   ENT
43 B02C 77                      LD     (HL),A           ; (HL)=A
44 B02D 23                      INC    HL               ; HL=HL+1
45 B02E 10FC                    DJNZ   LP03             ; B=B-1
46 B030                 ;                                 IF B<>0 GOTO LP03
47 B030 E1                      POP    HL               ; HL=(SP)
48 B031 19                      ADD    HL,DE            ; HL=HL+DE
49 B032 C1                      POP    BC               ; BC=(SP)
50 B033 10F4                    DJNZ   LP02             ; B=B-1
```

```
    **  Z80 ASSEMBLER SP-2102 PAGE 02  **

01 B035                 ;                                 IF B<>0 GOTO LP02
02 B035                 ;
03 B035 DBE8                    IN     A,(E8H)
04 B037 CBBF                    RES    7,A              ; V-RAM (ADR OFF)
05 B039 D3E8                    OUT    (E8H),A
06 B03B                 ;
07 B03B C9                      RET                     ; RETURN
08 B03C                 ;
09 B03C                         DEFS   04H
10 B040                 ;
11 B040                 X1:     ENT
12 B040 00                      DEFB   00H
13 B041                 Y1:     ENT
14 B041 00                      DEFB   00H
15 B042                 X2:     ENT
16 B042 00                      DEFB   00H
17 B043                 Y2:     ENT
18 B043 00                      DEFB   00H
19 B044                 Z1:     ENT
20 B044 00                      DEFB   00H
21 B045                 ;
22 B045                         END
```

```
    **  Z80 ASSEMBLER SP-2102 PAGE 01  **

01 0000                 ;
02 0000                 ;  PROGRAM 2
03 0000                 ;
04 0000                 ;       SORT
05 0000                 ;
06 0000                 ;            ALL Z-80 COMPUTER
07 0000                 ;
08 0000                         REL    B000H            ; ORG B000H
09 B000                 ;
10 B000 ED5B70B0                LD     DE,(DATAA)       ; DE=(DATAA)
11 B004 2170B0                  LD     HL,DATAA         ; HL=DATAA
12 B007 19                      ADD    HL,DE            ; HL=HL+DE
13 B008 19                      ADD    HL,DE            ; HL=HL+DE
14 B009 2270B0                  LD     (DATAA),HL       ; (DATAA)=HL
15 B00C                 LP100:  ENT
16 B00C 0172B0                  LD     BC,DATAB         ; BC=DATAB
17 B00F                 LP110:  ENT
18 B00F C5                      PUSH   BC               ; (SP)=BC
19 B010 DDE1                    POP    IX               ; IX=(SP)
20 B012                 LP120:  ENT
21 B012 C5                      PUSH   BC               ; (SP)=BC
22 B013 FDE1                    POP    IY               ; IY=(SP)
23 B015 FD23                    INC    IY               ; IY=IY+1
24 B017 FD23                    INC    IY               ; IY=IY+1
25 B019                 LP130:  ENT
26 B019 FD6601                  LD     H,(IY+01H)       ; H=(IY+01H)
27 B01C FD6E00                  LD     L,(IY+00H)       ; L=(IY+00H)
28 B01F DD5601                  LD     D,(IX+01H)       ; D=(IX+01H)
29 B022 DD5E00                  LD     E,(IX+00H)       ; E=(IX+00H)
```

```
30 B025 A7                    AND   A               ; CY=0
31 B026 ED52                  SBC   HL,DE           ; HL=HL-DE-CY=HL-DE
32 B028 3004                  JR    NC,LP150        ; IF CY=0 GOTO LP150
33 B02A             LP140:    ENT
34 B02A FDE5                  PUSH  IY              ; (SP)=IY
35 B02C DDE1                  POP   IX              ; IX=(SP)
36 B02E             LP150:    ENT
37 B02E FD23                  INC   IY              ; IY=IY+1
38 B030 FD23                  INC   IY              ; IY=IY+1
39 B032 FDE5                  PUSH  IY              ; (SP)=IY
40 B034 D1                    POP   DE              ; DE=(SP)
41 B035 2A70B0                LD    HL,(DATAA)      ; HL=(DATAA)
42 B038 A7                    AND   A               ; CY=0
43 B039 ED52                  SBC   HL,DE           ; HL=HL-DE-CY=HL-DE
44 B03B 30DC                  JR    NC,LP130        ; IF CY=0 GOTO LP130
45 B03D             LP160:    ENT
46 B03D FDE5                  PUSH  IY              ; (SP)=IY
47 B03F C5                    PUSH  BC              ; (SP)=BC
48 B040 FDE1                  POP   IY              ; IY=(SP)
49 B042 DD6601                LD    H,(IX+01H)      ; H=(IX+01H)
50 B045 DD6E00                LD    L,(IX+00H)      ; L=(IX+00H)
```

```
   **  Z80 ASSEMBLER SP-2102 PAGE 02  **

01 B048 FD5601                LD    D,(IY+01H)      ; D=(IY+01H)
02 B04B FD5E00                LD    E,(IY+00H)      ; E=(IY+00H)
03 B04E DD7201                LD    (IX+01H),D      ; (IX+01H)=D
04 B051 DD7300                LD    (IX+00H),E      ; (IX+00H)=E
05 B054 FD7401                LD    (IY+01H),H      ; (IY+01H)=H
06 B057 FD7500                LD    (IY+00H),L      ; (IY+00H)=L
07 B05A FDE1                  POP   IY              ; IY=(SP)
08 B05C             LP170:    ENT
09 B05C 03                    INC   BC              ; BC=BC+1
10 B05D 03                    INC   BC              ; BC=BC+1
11 B05E 2A70B0                LD    HL,(DATAA)      ; HL=(DATAA)
12 B061 2B                    DEC   HL              ; HL=HL-1
13 B062 2B                    DEC   HL              ; HL=HL-1
14 B063 A7                    AND   A               ; CY=0
15 B064 ED42                  SBC   HL,BC           ; HL=HL-BC-CY=HL-BC
16 B066 30A7                  JR    NC,LP110        ; IF CY=0 GOTO LP110
17 B068             LP180:    ENT
18 B068 C9                    RET                   ; RETURN
19 B069             ;
20 B069                       DEFS  07H
21 B070             ;
22 B070             DATAA:    ENT
23 B070 0000                  DEFW  0000H
24 B072             DATAB:    ENT
25 B072             ;
26 B072                       END
```

```
10 REM *************************
20 REM *                       *
30 REM *  PROGRAM 1            *
40 REM *                       *
50 REM *     BASIC    EXAMPLE  *
60 REM *                       *
70 REM *************************
80 REM
100 POKE &HB040,10,10,20,20,80
110 CALL &HB000
120 END
```

```
10 REM *************************
20 REM *                       *
30 REM *  PROGRAM 2            *
40 REM *                       *
50 REM *     BASIC    EXAMPLE  *
60 REM *                       *
70 REM *************************
80 REM
100 DIM A(100)
110 A(0)=100
120 FOR JA=1 TO A(0)
130   A(JA)=INT(RND(1)*256*256)
140 NEXT JA
150 FOR JA=1 TO A(0)
160   PRINT A(JA),
170 NEXT JA
180 PRINT:PRINT
200 REM **********
210 GOSUB 1000 :REM GOSUB 2000
220 REM **********
300 FOR JA=1 TO A(0)
310   PRINT A(JA),
320 NEXT JA
330 END
500 REM
510 REM =============================
520 REM
1000 REM
1010 REM ********** M.L. **********
1020 REM
1100 FOR JA=0 TO A(0)
1110   H=INT(A(JA)/256)
1120   L=A(JA)-H*256
1130   POKE &HB071+JA*2,H
1140   POKE &HB070+JA*2,L
1150 NEXT JA
1200 REM **********
1210   CALL &HB000
1300 REM **********
1310 FOR JA=1 TO A(0)
1320   H=PEEK(&HB071+JA*2)
1330   L=PEEK(&HB070+JA*2)
1340   A(JA)=H*256+L
1350 NEXT JA
1360 RETURN
2000 REM
2010 REM ********** BASIC **********
2020 REM
2100 FOR JA=1 TO A(0)-1
2110   XA=JA
2120   FOR JB=JA+1 TO A(0)
2130     IF A(XA)=<A(JB) GOTO 2150
2140     XA=JB
2150   NEXT JB
2160   SWAP A(XA),A(JA)
2170 NEXT JA
2180 RETURN
```

MZ-700/K/C/1200

モニタ・サブルーチンの解析とその使い方

K/C/1200と700の違い

西畑 文広

サブルーチンとは何か

マシン語プログラミングのサブルーチンを中心とした特集ということで，今回はサブルーチンの使い方を取りあげました。また，80K/C/1200との交換性という意味で モニター1Z-009Aを徹底的に解析し，なおかつSP-1002との比較をしましたので，マシン語のプログラムを作成するときには参考にしてください。

まず本文に入る前に，サブルーチンとは何なのかを説明しておかないと何を言っているのかわからなくなるので，ここで定義しておきます。

まとまりのある手順(命令語，その他)のかたまりをルーチンといい，このルーチンに対して，自分自身では単独に実行できないルーチンをサブルーチンといいます，またサブルーチンに対して，このサブルーチンを呼び出すもとのルーチンをメインルーチンといいます。

サブルーチンはプログラム中の共通的手続きを集めて作成しておく場合が多く，同じルーチンを何回も使用するときなど，1回そのサブルーチンを作れば，あとはそれを呼びだすだけですむというわけです。そしてそのプログラムのメモリ容量も少なくてすみます(図1)。

モニタ1Z-009Aの構成

そこで図2を参照してください。

① $0000～$0049番地までジャンプテーブルでおのおののサブルーチンのメニューになっています。

② $004A～$018F番地までがメインルーチンで，主にコマンドの解析になっています。

③ $018F～$0FFF番地までがサブルーチンの集まりです。

④ $1000～$11FF番地までがモニタのワークエリアです。

そして表1にこの1Z-009AとSP-1002をアドレス順に，サブルーチン単位で解析したので比較または参考にしてください。

図1

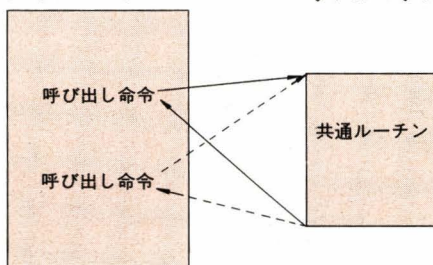
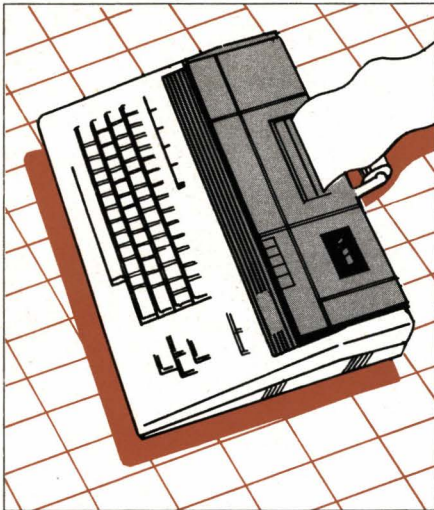



図2 モニタ1Z-009A

表1 比較表 △印はアドレス(飛び先)が違っていることを示します。

アドレス	1Z-009A			アドレス	SP-1002
	ジャンプ・テーブル				
0000	JP	START	(004AH)	→	→ (004AH)
		イニシャライズルーチンへ			
0003	JP	?GETL	(07E6H)	→	→ (07E6H)
		1行入力ルーチンへ			
0006	JP	?LTNL	(090EH)	→	→ (090EH)
		改行ルーチンへ			
0009	JP	?NL	(0918H)	→	→ (0918H)
		改行してない時改行ルーチンへ			
0000	JP	?PRTS	(0920H)	→	→ (0920H)
		スペースプリントルーチンへ			
000F	JP	?PRTT	(0924H)	→	→ (0926H)△
		10カラムずつそろえるルーチンへ			
0012	JP	?PRNT	(0935H)	→	→ (0935H)
		1文字プリントルーチンへ			
0015	JP	?MSG	(0893H)	→	→ (0981H)△
		メッセージプリントルーチンへ(カーソルコントロール)			
0018	JP	?MSGX	(08A1H)	→	→ (0999H)△
		メッセージプリントルーチンへ			
001B	JP	?GET	(08BDH)	→	→ (08BDH)
		1文字入力ルーチンへ			
001E	JP	?BRK	(0A32H)	→	→ (0A32H)
		ブレイクキー入力ルーチンへ			
0021	JP	?WRI	(0436H)	→	→ (0436H)
		ライト情報ルーチンへ			
0024	JP	?WRD	(0475H)	→	→ (0475H)
		ライトデータルーチンへ			
0027	JP	?RDI	(04D8H)	→	→ (04D8H)
		リード情報ルーチンへ			
002A	JP	?RDD	(04F8H)	→	→ (04F8H)
		リードデータルーチンへ			
002D	JP	?VRFY	(0588H)	→	→ (0588H)
		データ比較ルーチンへ			

0030	JP	?MLDY メロディー演奏ルーチンへ	(01C7H)	→	→	(01C7H)
0033	JP	?TMST タイムセットルーチンへ	(0308H)	→	→	(0308H)
0036	NOP	割り込みのアドレスを合せるため				
0037	NOP					
0038	JP	1038H 割り込み処理ルーチンへ				
003B	JP	?TMRD タイムリードルーチンへ	(0358H)	→	→	(0358H)
003E	JP	?BEL ベルオンルーチンへ	(0577H)	→	→	(02E5H)△
0041	JP	?TEMP テンポセットルーチンへ	(02E5H)	→	→	(02FAH)△
0044	JP	MLDST メロディースタートルーチンへ	(02ABH)	→	→	(02ABH)
0047	JP	MLDSP メロディーストップルーチンへ	(02BEH)	→	→	(02BEH)

比較表

1Z-009A

アドレス	メインルーチン
004A	START：　(コールドスタート) マシンイニシャライズ（初期設定）
00AD	ST1：　(ホットスタート) コマンド解析をして，それぞれに制御を移す。
00EA	GOTO： Jコマンドの時に来るルーチンで，Jの後の数字のアドレスに制御を移す。
00EE	SG： Bコマンドの時に来るルーチンで，キーサウンドのON，OFFを制御する。
00F6	FD： Fコマンドの時に来るルーチンで，フロッピー制御ROMが付いていればそちらに制御を移す。
00FE	?ER： エラー解析ルーチン
0108	LOAD： Lコマンドの時に来るルーチンで，テープよりプログラムをロードし，そのプログラムの実行アドレスに制御を移す。
0126	BGETL： 1行入力し，その時ブレイクキーをチェックして，ブレイクの時ホットスタートにもどる。
0134	HEXY： DEの内容の先頭アドレスから連続した4個のアスキー列を16進数と見なし，HLレジスターにセットしてリターン。但し16進数でない時はホットスタートにもどる。
013E	MSG?2： "LOADING", 0DH アスキー列が格納されている。
0147	MSGE1： "CHECK SUM ER", 0DH アスキー列が格納されている。
0155	PTEST： Pコマンドの処理ルーチンでPの後になんの文字があるか解析し，それぞれに制御を移す。
0170	PTST1： Pコマンドの後が&でない時来るルーチンでプリンターに1行印字してホットスタートにもどる。
0176	LPT： P&Lの処理ルーチンで，文字桁数を40文字/行に変更する。 ホットスタートにもどる。
0178	LPT： P&Sの処理ルーチン 文字桁数を80文字/行に変更する。 ホットスタートにもどる。
0180	PTRN： P&Tの処理ルーチン テスト・パターン印字 PTSTOにもどる
0184	PLOT： P&Gの処理ルーチン グラフィックモードにする。 PTSTOにもどる
018B	PEN： P&Cの処理ルーチン ペンを次の色に変える PTSTOにもどる。
アドレス	**サブルーチン**
018F	LPRNT： Aレジにあるデータをアスキコードと見て1文字プリンタにプリントアウトする。
01A5	PMSG： DEレジスタの内容をメッセージの先頭アドレスと見なし，そこからアスキコード列をプリンタに0Dコードがあるまでプリントアウトする。

SP-1002

アドレス	メインルーチン
004A	START：　(コールドスタート) マシンイニシャライズ（初期設定）
0082	ホットスタート コマンド解析し，それぞれに制御を移す。
00CF	LOAD： LOADコマンド処理ルーチン
0131	'FOUND', 'LOADING', '**MONITOR SP-1002 **' のメッセージデータ
0158	GOTO： GOTOコマンド処理ルーチン
016B	SS： SSコマンド処理ルーチン
0173	SG： SGコマンド処理ルーチン
0177	FD： FDコマンド処理ルーチン
アドレス	**サブルーチン**
0180	一致検出処理ルーチン
0196	コマンド用比較用データ 'LOAD', 'GOTO', 'SS', 'SG', 'FD'
01A4	エラー処理ルーチン
01B5	"CHECK SUM ERROR" メッセージデータ

01C7	?MLDY: DEレジスタの内容を音楽データの先頭アドレスと見なし，そこからアスキコード列を0D又C8があるまで演奏する。	01C7	?MLDY: 1Z-009Aに同じ
02A6	4DE: DEレジスタを4回インクリメントする DE←DE+4		
02AB	MLDST: $111A,$11A2の内容を分周比と見なし，その周波数を音を鳴らしつづける。	02AB	MLDST: 1Z-009Aに同じ
02BE	MLDSP: 音の発生を止める。	02BE	MLDSP: 1Z-009Aに同じ
02C8	RYTHM: ブレイクキーのチェックとBレジスタの内容を音符の長さと見なし，その回数時間待ちをする。	02C8	RYTHM: 1Z-009Aに同じ
02E5	?TEMP: Aレジスタの内容をテンポの1～7と見なし，演奏テンポを設定する。	02E5	?BEL: ベルを一瞬ならす。
02F3	MANG: カーソルの位置を計算する。	02FA	?TEMP: 1Z-009Aに同じ
0308	?TMST: AレジスタとDEレジスタの内容をそれぞれAM，PMの区別に0と1とDEに時刻（秒）であると見なし，内蔵されている時計をセットし，起動する。	0308	?TMST: 1Z-009Aに同じ
0358	?TMRD: 内蔵されている時計の値を読み取り，AレジスタにAM，PMの区別に0と1，又DEレジスタに時刻(秒)をセットしてリタンする。	0358	?TMRD: 1Z-009Aに同じ
038D	TIMIN: 割り込みが発生した時の処理ルーチン 12時間を秒数に直した値をカウンタにセットし，AM，PMを変換する。	0392	TIMIN: 1Z-009Aに同じ
03B1	SPHEX: スペースをプリントし，HLレジスタの内容のアドレスの示すデータをHEXと見なし2文字出力する。		
03BA	PRTHL: HLレジスタの内容を4ケタのHEXコードと見なし，4文字プリントする。	03BA	PRTHL: 1Z-009Aに同じ
03C3	PRTHX: Aレジスタの内容を2ケタのHEXコードと見なし，2文字プリントする。	03C3	PRTHX: 1Z-009Aに同じ
03D3	SLPT: 文字桁数を80文字／行に変更するデータ（プリンタ）		
03DA	ASC: Aレジスタの内容をHEXコードと見なし，アスキーコードに変換する。	03DA	ASC: 1Z-009Aに同じ
03E5	HEXJ: Aレジスタの内容をアスキーコードと見なし，HEXコードに変換する。	03E9	HEXコードとアスキーコードの変換表
		03F9	HEXJ: 1Z-009Aに同じ
03FB	MSG#1: 'PLAY,' 'RECORD.' のメッセージデータ。		

0410	HLHEX: DEレジスタの内容のアドレスから連続した4個のアスキ列を16進数列と見なし，アスキコードをHEXコードに変換し，HLレジスタにセットする。	0410	HLHEX: 1Z-009Aに同じ
041F	2XEX: DEレジスタの内容のアドレスから連続した2個のアスキ列を16進数列と見なしアスキコードをHEXコードに変換し，Aレジスタにセットする。	041F	2XEX: 1Z-009Aに同じ
0436	?WRI: テープにヘッダ情報をライトする。	0436	?WRI: 1Z-009Aに同じ
0467	MSG#7: 'WRITING' のメッセージデータ	046C	MSG#7: 1Z-009Aに同じ
0470	LLPT: 文字桁数を40文字／行に変更するデータ（プリンタ）		
0475	?WRD: テープにデータ部をライトする。	0475	?WRD: 1Z-009Aに同じ
048A	WTAPE: テープへ実際に書き込むルーチン		WTAPE: 1Z-009Aに同じ
04D8	?RDI: テープからヘッダ情報をリードする。	04D8	?RDI: 1Z-009Aに同じ
04F8	?RDD: テープからデータ部をリードする。	04F8	?RDD: 1Z-009Aに同じ
050E	RTAPE: テープから実際に読み込むルーチン	0510	RTAPE: 1Z-009Aに同じ
0577	?BEL: ベルを一瞬ならす。		
057E	FLKEY: カーソル点滅をし，1文字Aレジスターに GETする。		
0588	?VRFY: テープのデータ部の内容とメモリデータを比較する。	0588	?VRFY: 1Z-009Aに同じ
05AD	TVRFY: テープデータとメモリデータを実際に比較するルーチン。	05B2	TVRFY: 1Z-009Aに同じ
05F0	?LOAD: カーソル点滅キー入力の時使用するルーチンで，そこに本来あるべき文字をディスプレイする。		
05FA	NLPHL: 一行改行して，HLレジスターの内容を4ケタのHEXコードと見なし4文字プリントする。		
0601	EDGE: テープの読み込み時のスタートビットの同時を取る。	0601	EDGE: 1Z-009Aに同じ
0624	RBYTE: テープから1バイト読み込む。	0624	RBYTE: 1Z-009Aに同じ
065B	TMARK: テープのリーダー音を待ち合わせる。	065E	TMARK: 1Z-009Aに同じ
069F	MOTOR: テープレコーダのモータをスタートさ	06B2	MOTOR: 1Z-009Aに同じ

↗

	テープレコーダのモータをスタートさせる。		
06E7	MSG?3: '** MONITOR 1Z-009A **'のメッセージデータ		
0700	MSTOP: テープレコーダのモータをストップさせる	0700	MSTOP: 1Z-009Aに同じ
071A	CKSUM: チェックサム用のデータを作成する。	0722	'↓PLAY', '↓RECORD' メッセージデータ
073E	?MODE: キーポイント(E003)のモードを初期設定する。	0733	CKSUM: 1Z-009Aに同じ
0748	?KY3: ?KEYルーチンで使用されるルーチンで小文字の処理をする。		
0759	DLY1: 107 usディレイ	0759	DLY1: 1Z-009Aに同じ
0760	DLY2: 99 usディレイ	0760	DLY2: 1Z-009Aに同じ
0767	?KY6: ?KEYルーチンで使用されるルーチン	0767	0を1ビット書き込むルーチン
076D	WBYE: テープに1バイト書き込みをする。		
0780	GAP: テープにリーダー音を書き込む。	0780	1を1ビット書き込むルーチン
07AE	MCOR: Mコマンドの処理ルーチンでサブルーチンでまない。	07A5	WBYE: 1Z-009Aに同じ
		07B8	GAP: 1Z-009Aに同じ
07E6	?GETL: キーボードから一行分をDEレジスタの示すアドレスに入力する。	07E6	?GETL: 1Z-009Aに同じ
0893	?MSG: DEレジスタの内容をメッセージの先頭アドレスと見なし，そこからアスキコード列をディスプレイに0Dコードがあるまで出力する。		
08A1	?MSGX: ?MSGルーチンと同様にメッセージをディスプレイするが，カーソルコントロールの処理をせず，そのままコントロールコードもディスプレイする。		
08BD	?GET: キーボードから1文字だけアスキーコードでAレジスタに取り込む。	08BD	?GET: 1Z-009Aに同じ
08CA	?KEY: キーボードから1文字だけディスプレイコードでAレジスタに取り込む。	08CA	?KEY: 1Z-009Aに同じ
090E	?LTNL: 常に改行する。	090E	?LTNL: 1Z-009Aに同じ
0918	?NL: 改行されてない時だけ改行する。	0918	?NL: 1Z-009Aに同じ
		0920	

0924	?PRTT: BASICのTAB(10) 処理ルーチン10カラムづつにそろえる		スペースをプリントするルーチン
		0926	?PRNT: 1Z-009Aに同じ
0935	?PRNT: Aレジスタの内容をアスキーコードと見なし，1文字ディスプレイする。	0935	?PRT: 1Z-009Aに同じ
0942	MSGOK: 'OK!'のメッセージデータ		
0946	?PRT: Cレジスタの内容をアスキーコードと見なし，1文字ディスプレイする。	0946	?PRT: 1Z-009Aに同じ
097B	GETLH: ?GETLルーチンで使用されているルーチンで，ひらがなモードにする。	0970	ディスプレイコードによる1文字プリントルーチン
0980	#BRK: ?KEYルーチンで使用されるルーチンでブレクキーの処理をする。	0981	メッセージプリントルーチン
0986	SMALL: L←L－22H		
098B	FLASI: ?FLSルーチンで使用されるルーチン。		
0990	?BRKZ: ?BRKルーチンで使用されるルーチン。		
099B	MSGSV: 'FILENAME?' メッセージデータ	0999	メッセージプリントルーチン
09A6	DLYR: 7msのデイレイ		
09B3	?KEY: カーソルを点滅させながらキーボードから1文字だけディスプレイコードでAレジスターに取り込む。	09B3	?KEY: 1Z-009Aに同じ
09DD	#CLR08: HLレジスタにクリヤ先頭アドレスがセットされていると見なしメモリを2Kバイトクリアする。		
09FF	?FLAS: カーソルの点滅をする。	09FF	?FLAS: 1Z-009Aに同じ
0A01	?KYK: ?KEYルーチンで使用されるルーチンでカナ処理をする。		
0A32	?BRK: シフトブレイクが押されているかのチェック。	0A32	?BRK: 1Z-009Aに同じ
0A4A	DLY3: 320 usのディレイ		
0A50	?SWEP: キーボードをスキャンし，Bレジスタにステイタス，Cレジスタにキーコードをセットする。	0A50	?SWEP 1Z-009Aに同じ
0A92	ATBL: アスキーコードとディスプレイコードの変換表。		
0B92	?SAVE: カーソルのマークをディスプレイする。		

0BB9	?ADCN： Aレジスタの内容をアスキコードと見なし，ディスプレイコードに変換する。	0BB9	?ADCN： 1Z-009Aに同じ
0BC5	VRNS： 'V1. 0A' バーションタ名		
0BCE	?DACN： Aレジスタの内容をディスプレイコード見なし，アスキコードに変換する。	0BCE	?DACN： 1Z-009Aに同じ
0BEA	KTBL： キーマトリックコードとディスプレイコードの変換表。	0BD6	アスキーコードとディスプレイコードの変換表。
		0CC6	ディスプレイコードとアスキーコードの変換表。
0D47	SHORT： テープに0を1ビット書き込む。		
0D60	LONG： テープに1を1ビット書き込む。		
0DA6	?BLNK： 垂直ブランクを待ち合せる。	0DA6	?BLNK：左に同じ ↗
0DB5	?DSP： Aレジスタの内容をディスプレイコードと見なし表示する。	0DB5	?DSP： 1Z-009Aに同じ
0DDC	?DPCT： Aレジスタの内容をディスプレイコントロールコードと見なしディスプレイをコントロールする。	0DDC	?DPCT： 1Z-009Aに同じ
0F58	SAVE： Sコマンドの処理ルーチンでサブルーチンではない。		
0FB1	?POLT： カーソル位置をHLレジスタにアドレスとしてセットする。	0FB1	?POLT： 1Z-009Aに同じ
		0FC9	I/Oポートイニシャライズルーチン
0FCB	VRFY： Vコマンドの処理ルーチンでサブルーチンではない。	0FD8	メモリクリアルーチン

サブルーチンの使い方

たとえば表1のサブルーチン・メニューの中から'MSG1'のメッセージ出力ルーチンを使ってみましょう。

なお，この表で使っている「SG」「ST1」などのラベル名は，MZ-700のOWNER'S MANUALと合わせてあります。

このルーチンの前提条件は，DEレジスタに出力したいメッセージのアスキーコード列の先頭アドレスをセットします。そしてそのアドレスから出力したいメッセージを入れ，メッセージの最後に16数の0Dを入れます。それでは出力したいメッセージをTESTとし，メッセージの先頭アドレスを$A009から入れます。そしてプログラムを$A000から入れます(図3)。

最初にDEレジスタにメッセージ先頭アドレスを入れます。そしてモニターのMSG1のサブルーチンをコールします。最後に，

図3

アドレス	内容	
$A000	11	DE＝$A009
A001	09	
A002	A0	
A003	CD	CALL $0015
A004	15	
A005	00	
A006	C3	JP $00AD
A007	AD	
A008	00	
A009	54	T
A00A	45	E
A00B	53	S
A00C	54	T
A00D	0D	エンドマーク

このまま動かしては爆走してしまうのでモニタのホットスタートである$00ADにジャンプします。それではMコマンドで下記のように実際に打ち込んでください。

```
*MA000     CR
A000   11  CR
A001   09  CR
A002   A0  CR
A003   CD  CR
A004   15  CR
A005   00  CR
A006   C3  CR
A007   AD  CR
A008   00  CR
A009   54  CR
A00A   45  CR
A00B   53  CR
A00C   54  CR
A00D   0D  CR
```

すべて入れ終わりましたか？　さて，打ち込みが終わったら下記のようにJコマンドでジャンプして動かします。

```
*JA000  CR
```

ディスプレイにTESTと表示されましたか？　それではついでに，メッセージをプリンタにプリントしてみましょう。プリンタを持っていない人はここは少し休んでいてください。

プリンタにプリントするサブルーチンはは$01A5のアドレスのPMSGを使います。このサブルーチンは先ほどのMSG1と使い方は同じで，DEレジスタにセットとしてコーすればよいのです。またMコマンドで下記のようにコールのアドレスを変更するだけですみます。

```
A004  15   A5  CR
A005  00   01  CR
```

それでは先ほどと同じように，Jコマンドで$A000にジャンプして動かしてみてください。プリンタにTEST出力されていればOKです。

このようにマシン語でサブルーチンをコールするプログラムを作成すれば，サブルーチンを自由に使いこなすことができますので，ためしてください。

互換性について

最後にBASICプログラムの交換性ですが，SP-5030で作成されているプログラムはS-BASICでそのまま読ませれば，S-BASICがコンバータをもっているので変換して動きます。が，一部変換されないものがあるので，できればSP-5030を購入してMZ-700でSP-5030をロードしてからBASICプログラムをロードして動かした方が無難でしょう。

しかしSP-5030でも動かないときがあります。たとえばそのBASICプログラムの中でマシン語を使用してモニタのジャンプテーブル以外のサブルーチンをコールしているプログラムは，動かないものがあります。これについては4月号で紹介したSP-1002のコンパチモニタのFN-700を使用すれば解決がつきます。

それでも動かないプログラムは，プログララム変更するしかありません。

MZ-700 (Hu-BASIC)

BASICというジャングルへ PARTII

逆アセンブラにハードコピーを

青木 孝

1. 使ってもらえたでしょうか，逆アセンブラ

Oh! MZ 5月号の逆アセンブラは，必要最小限のコマンドしか用意されていないので，使っているうちにいろいろと不満も出てきたのではないでしょうか。たとえば，必ず22行表示して止まるから，ある番地からある番地までプリンタで出力したい，というときに何度もキーを押さなければならないので面倒だなあ，といった，不満を持ったらすぐに改造にとりかかるとよいと思います。BASICで書いてあるのでやりやすいでしょう。というわけで，遅まきながら逆アセンブラ（5月号）を改造する手助けとなる変数の意味一欄表を示します(表1)。

表1

ADDR	：逆アセンブルする機械語のある番地を示す。
LNS	：逆アセンブルする行数を示す。行番号70で22にセットされている。
R$	：コマンドが入っている。INKEY$(2)で読み込んでいるので，R$には1文字しか入らない。従ってコマンドは1文字(N,M等)。
AD I	：1行の先頭番地を示す。

行番号190からがメインのサブルーチン（何だか変ですが）です。ADDRに逆アセンブルする番地をセットして呼ぶと，M$にメモリの内容を16進数で，NM$+OPER$にニモニックとオペランドを代入して戻ってきます。行番号1020からのサブルーチンは出力ルーチンです。

Hu-BASICにはラベルをつける機能があるのに，なぜそれを使ってプログラムを読みやすくしないのか，と思った人はいませんか。最初はラベルをたくさん使って書いてあったのです。ところが，速度が遅い。もしかしたら，と思いテストしてみたところ，ラベルは毎回サーチを行なってジャンプしていることが解りました。GOTO"行番号"とすると，Hu-BASICは，1回目の実行時に"行番号"を実際にその行が入っているメモリ番地に書き換えます。だからプログラムの最初の方へGOTOしようと，後の方へGOTOしようと，必要な時間は変わりないのです。それに対してGOTO"ラベル"とするとプログラムの最初から順に"ラベル"を探し，見つかったらそこへGOTOする，というような方法をとっているようです。だからプログラムの最後の方へGOTOする方が，最初の方へGOTOするより時間がかかることになるのです。他の機種のHu-BASICでは事情が異なっているかも知れませんが，MZ-700のHu-BASICでラベルを使うと実行速度が行番号を使った時よりも遅くなります。読みやすさを考えればラベルを使う方が良いのですが，困ったものです。

2. ハードコピーをつけたいのです

マルチスクリーン（5月号参照）を見て，Apple社のLisaを想い浮べた人はまずいないと思いますが，実はLisaやStarなどのいわゆるスーパーパーソナルコンピュータが実現しているマルチウィンドウを意識して，ほんの少しだけ（本当にほんの少しです）マルチウィンドウの使い心地を体験しようと思ったのがマルチスクリーンだったのです。

2枚のスクリーンではペーパーレスで仕事をするわけにはいきません。そこぜ今回は，画面をそのままプリンタに出力するハードコピーを作成し，ふたたびBASICに組み込んでしまいたいと思います。

ハードコピーができるようになれば，2枚で足りなくなったときには，1枚あるいは2枚ともハードコピーをとって，新しい部分を逆アセンブルしてハードコピーと一緒に解読を進めればよくなります。紙とインクがある限り何枚もスクリーンがあるようなものです。

3. 作ってみましょう

最初から機械語（アセンブラ）で書き下してもよいのですが，全体のアルゴリズムをフローチャート(最近は流行りませんね)に書くつもりでBASICでプログラムを書いて，それを見ながら機械語に書き直してみましょう。慣れてくればこんな面倒なことはしませんけれど……。

リスト1

```
LIST
10 FOR Y=0 TO 24
20 FOR X=0 TO 39
30 A$=CHARACTER$(X,Y)
40 LPRINT A$;
50 NEXT X
60 LPRINT
70 NEXT Y
80 END
OK
RUN
```

要は画面の文字を左上隅から横方向に順に調べてその文字をプリンタに出力すればよいのですからリスト1のようなプログラムになるのではないでしょうか。このリストもこのプログラムを実行することで出力していますから，LISTやRUNも書かれるのです。

非常に簡単なプログラムですから機械語にすぐなると思ってはいけません。問題はCHARACTER$(X,Y)とLPRINTなのです。まずA$には1文字しか入りませんからこの変数はAレジスタ(アキュムレータ)と対応させましょう。YとXにはBレジスタとCレジスタを対応させます。

次にCHARACTER$(X,Y)を機械語にします。CHRACTER$(X,Y)は画面の(X,Y)にある文字を読み込む関数ですから，(X,Y)がV-RAMの何番地になるのかを計算してLD(ニモニックの1つです)命令で Aレジスタに V-RAMの内容を読み込みます。V-RAMはD000番地から始まっているので，(X,Y)

の番地は,

(X,Y)の番地＝$D000＋Y×40＋X

で表わせます(40は10進数です)。

ところがZ-80は乗数命令を持ち合わせていないので40倍などは数命令に展開するか,サブルーチンをBASICの中から探し出してそこを呼び出さねばなりません。面倒ですね。この場合は,もっと簡単な方法があるので,この方法はやめにします。

V-RAMの番地は画面の左上隅が$D000で左へ順に1ずつふえていっていますから,HLレジスタに$D000をまず入れておいて,ひとつ読み出すごとにHLレジスタの内容を1ふやせばよいのです。

したがってA$=CHARACTER$(X,Y)は,

```
LD    A,(HL)
INC   HL
```

と書けます。

LPRINT A;にとりかかりましょう。実はCHARACTER$でもLPRINTでもBASICの処理ルーチンの番地は解っているのですから(5月号をみてください),そこをCALLすれば良いのです。しかし,引数(LPRINTのA$にあたるもの)の渡し方が良く解っていないので(探検不足だ!という批判の声が聞こえるようですが,ひたすら無視),もう少し下のレベルのサブルーチン,つまり1文字,プリンタへ出力するサブルーチンを探します。

例によってMZ-700OWNER'S MANUALのP.222を開きプリンタコマンドPの処理ルーチンの番地を探し,やがてP.224の$18Fのサブルーチンを見つけ出します。この中のOUT命令が重要です。OUT␣(FFH),AにあたるP3␣FF␣それに続く3E␣80␣D3␣FEをHu-BASICのFコマンドで探し,見つかった付近を逆アセンブルして,それらしいルーチンを見つけました。$1821からがそれです。AレジスタにASCIIコードを入れ,CALL $1821を実行すれば(機械語で)1文字プリンタに出力されます。したがってLPRINT A$:は

```
CALL  $1821
```

と書けます。

リスト2

```
CF00   D3E3      OUT      ($E3),A
CF02   E5        PUSH     HL
CF03   C5        PUSH     BC
CF04   2100D0    LD       HL,$D000
CF07   0619      LD       B,$19
CF09   0E28      LD       C,$28
CF0B   7E        LD       A,(HL)
CF0C   CDC005    CALL     $05C0
CF0F   CD2118    CALL     $1821
CF12   23        INC      HL
CF13   0D        DEC      C
CF14   79        LD       A,C
CF15   B7        OR       A
CF16   20F3      JR NZ    $CF0B
CF18   05        DEC      B
CF19   78        LD       A,B
CF1A   B7        OR       A
CF1B   20EC      JR NZ    $CF09
CF1D   3E0D      LD       A,$0D
CF1F   CD2118    CALL     $1821
CF22   C1        POP      BC
CF23   E1        POP      HL
CF24   D3E1      OUT      ($E1),A
CF26   C9        RET
```

V-RAMを読み出すには

```
OUT   ($E3),A
```

を実行することを忘れないようにして,さらにHLレジスタとBCレジスタの値をこわさないように

```
PUSH   HL
PUSH   BC
```

を最初に実行します。

あと1つ問題があります。V-RAMを読んだ値はASCIIコードではなくディスプレイコードだ,という点です。ディスプレイコードをASCIIコードに変換しなければなりません。モニタ1Z-009AにはP.207にあるように,?DACNというサブルーチンが$BCE番地にあります。これにあたるサブルーチンをHu-BASICでも探しました。その結果,ディスプレイコードをASCIIコードに直すのは$5C0番地に,逆にASCIIコードをディスプレイコードに直すのは$5D7番地にあることが解りました。

以上を合わせて作ったプログラムがリスト2です。このリストでは$CF00番地から始まっていますが,$CFFF番地より低い番地であればどこに配置しても動きます(このようなプログラムのことを,リロケータブルなプログラム,と呼びます)。ただし当然のことながらBASICと重なるところに置くことはできません。

4. まだまだ満足できないのです

逆アセンブラの出力をハードコピーする場合にはリスト2のプログラムで正しくハードコピーできますが,画面上にグラフィック文字があると,プロッタプリンタでは色を変え,16進数を打ち出してしまい行数がずれます。第1の改良点はグラフィック文字のときに,スペースを出力するようにすることです。

もう1つの不満はブレークキーが使えないので全画面を出力し終わるまで待たなければいけないことです。

5. 不満解消への道程

プロッタプリンタが,色を変え,16進数を打ち出すのは,マニュアルのP.213に書いてある文字以外は打ち出せないからです。だから,グラフィック文字も打ち出せるプリンタを周辺機器としてつないでいる人は正しく(そのプリンタの文字が)打ち出されるはずです。

Hu-BASICでは，リストをプロッタに出力すると，グラフィック文字等が．(ピリオド）になって出力されます。つまり，無い文字を．に変換しているルーチンがあると考えられます。ハードコピーのためだけならば，このルーチンを探すより，自分で比較を行なって，無い文字のASCIIコードがAレジスタに入っているときには，スペース($20）に変えてプロッタへ出力する方が，プログラムは長くなりますが，簡単です。ところが，普通のプリンタを付けていても，リストをとると，グラフィック文字をプリンタは打ち出せるにもかかわらず，．になってしまうという問題がありました（5月号，ごめんなさいのページ参照）ので，その解決の意味もあって，．に変換しているルーチンを探しました。リスト3がそのルーチンです．下線部が．のASCIIコード$2EをAレジスタに入れているところです。ついでながら逆アセンブラの限界がこのリストに表われています。それは，線で囲んだ部分です。リストを読むとこの部分はデータになっていることが解ります。ところが逆アセンブラは何も考えずに機械語だと思ってニモニックとオペランドを出力しています。逆アセンブラリストを読んでいて変なニモニックに行きあたったら，データではないか，と考えてみるのがよいでしょう。

さて，普通のプリンタに正しくリストを出すには，下線部の2バイトを00_00に書き換えてNOPにしてしまえばよいでしょう。

プロッタに出力するときには，$17ECを，$2Eから$20に変えて$17D2をCALLすればよいのです。ただしこのルーチンはπを出力せず，．に変えてしまいます。気になる人は自分でサブルーチンを作ってください。

次にブレークキーが使えるようにします。ブレークするのは1行打ち終ったときにしましょう。これもきっとサブルーチンが存

リスト3

```
17D2  FE20    CP      $20
17D4  381C    JR C    $17F2
17D6  FE60    CP      $60
17D8  3847    JR C    $1821
17DA  FE70    CP      $70
17DC  380C    JR C    $17EA
17DE  FED7    CP      $D7
17E0  283F    JR Z    $1821
17E2  FECF    CP      $CF
17E4  283B    JR Z    $1821
17E6  FEC1    CP      $C1
17E8  3837    JR C    $1821
17EA  F5      PUSH    AF
17EB  3E2E    LD      A,$2E
17ED  CD2118  CALL    $1821
17F0  F1      POP     AF
17F1  C9      RET
17F2  F5      PUSH    AF
17F3  E5      PUSH    HL
17F4  C5      PUSH    BC
17F5  0606    LD      B,$06
17F7  210E18  LD      HL,$180E
17FA  BE      CP      (HL)
17FB  23      INC     HL
17FC  280A    JR Z    $1808
17FE  23      INC     HL
17FF  10F9    DJNZ    $17FA
1801  CDEA17  CALL    $17EA
1804  C1      POP     BC
1805  E1      POP     HL
1806  F1      POP     AF
1807  C9      RET
1808  7E      LD      A,(HL)
1809  CD2118  CALL    $1821
180C  18F6    JR      $1804
180E  1F      RRC
180F  111E12  LD      DE,$121E
1812  1C      INC     E
1813  13      INC     DE
1814  1D      DEC     E
1815  14      INC     D
1816  0B      DEC     BC
1817  15      DEC     D
1818  0C      INC     C
1819  163E    LD      D,$3E
181B  FF      RST 38
```

リスト4

```
89B9  E5      PUSH    HL
89BA  C5      PUSH    BC
89BB  213E20  LD      HL,$203E
89BE  22EB17  LD      ($17EB),HL
89C1  2100D0  LD      HL,$D000
89C4  0619    LD      B,$19
89C6  0E28    LD      C,$28
89C8  D3E3    OUT     ($E3),A
89CA  7E      LD      A,(HL)
89CB  D3E1    OUT     ($E1),A
89CD  CDC005  CALL    $05C0
89D0  CDD217  CALL    $17D2
89D3  23      INC     HL
89D4  0D      DEC     C
89D5  79      LD      A,C
89D6  B7      OR      A
89D7  20EF    JR NZ   $89C8
89D9  CD1F18  CALL    $181F
89DC  D3E3    OUT     ($E3),A
89DE  3EF8    LD      A,$F8
89E0  3200E0  LD      ($E000),A
89E3  3A01E0  LD      A,($E001)
89E6  07      RLCA
89E7  3005    JR NC   $89EE
89E9  05      DEC     B
89EA  78      LD      A,B
89EB  B7      OR      A
89EC  20D8    JR NZ   $89C6
89EE  3A01E0  LD      A,($E001)
89F1  07      RLCA
89F2  30FA    JR NC   $89EE
89F4  D3E1    OUT     ($E1),A
89F6  213E2E  LD      HL,$2E3E
89F9  22EB17  LD      ($17EB),HL
89FC  C1      POP     BC
89FD  E1      POP     HL
89FE  C9      RET
```

在するに違いないのですが，見つけられなかったので自分で組みました。

MZ-700のキーボード・スキャンの方法はOh! MZ誌上で何度かとりあげられたようなので，原理の説明などはそちらを読んでください。やり方だけを示します。マニュアルのP.196にあるキーボードマトリックス回路図を見てください。押されたかどうかを知りたいキーを探します。ここではBREAKですから，⑨と⑪の交わるところにあります。列の数字(この場合9)から1を引いて$F0を加えた数を$E000番地にロードします。つまり，ブレークキーなら

```
LD   ($E000),$F8
```

をまず実行します。次に$E001番地の内容を読み込みます。つまり

```
LD   A,($E001)
```

とするのです。そして行にあたるビット，ブレークキーなら⑪つまりD_7，1番上のビットが0か1かを調べ，0なら押されている1なら押されていない，と判断するのです。さらに例をあげると，

```
LD   ($E000),$F3
LD   A,($E001)
```

を実行し，Aレジスタに$B7が入っていたとすると，$B7は2進数で10110111ですからD_6ビットとD_3ビットが0，すなわち，JとMの2つのキーが押されているということが解るのです。ブレークキーが押されている，と解ったらループを抜け出すのですが，抜け出たら今度はブレークキーが押されていない状態になるまで待たねばなりません。そうしないとたとえば逆アセンブラではコマンドとしてブレークキーが押された，ということになってしまうのです。

以上の2点を組み込んだハードコピーのプログラムがリスト4です。開始番地が，$89B9になっているのはBASICのコマンドにしたからです。

リスト5

```
89B9   E5          PUSH    HL
89BA   C5          PUSH    BC
89BB   210000      LD      HL,$0000
89BE   22EB17      LD      ($17EB),HL
89C1   2100D0      LD      HL,$D000
89C4   0619        LD      B,$19
89C6   0E28        LD      C,$28
89C8   D3E3        OUT     ($E3),A
89CA   7E          LD      A,(HL)
89CB   D3E1        OUT     ($E1),A
89CD   CDC005      CALL    $05C0
89D0   CDD217      CALL    $17D2
89D3   23          INC     HL
89D4   0D          DEC     C
89D5   79          LD      A,C
89D6   B7          OR      A
89D7   20EF        JR NZ   $89C8
89D9   CD1F18      CALL    $181F
89DC   D3E3        OUT     ($E3),A
89DE   3EF8        LD      A,$F8
89E0   3200E0      LD      ($E000),A
89E3   3A01E0      LD      A,($E001)
89E6   07          RLCA
89E7   3005        JR NC   $89EE
89E9   05          DEC     B
89EA   78          LD      A,B
89EB   B7          OR      A
89EC   20D8        JR NZ   $89C6
89EE   3A01E0      LD      A,($E001)
89F1   07          RLCA
89F2   30FA        JR NC   $89EE
89F4   D3E1        OUT     ($E1),A
89F6   213E2E      LD      HL,$2E3E
89F9   22EB17      LD      ($17EB),HL
89FC   C1          POP     BC
89FD   E1          POP     HL
89FE   C9          RET
```

6. コマンドとして登録しましょう

新しいBASICのテープを作るのであれば5月号の変更点に加えて，$9BB9からリスト4を入力します。コマンド名は"HRDC"として，$40E8から48␣52␣44␣C3とします。実行番地は$43ECからB9␣89です。これだけの変更を加えて，テープへセーブしてください。そのテープからロードしたBASICはHRDC (CR) でプロッタ（あるいはプリンタ）が動きだし，ブレークキーを押しておくと，1行終ったところで印刷をやめるはずです。

グラフィック文字を印刷できるプリンタをつないでいる人はリスト4ではなくリスト5を使ってください。

いずれにせよ，テープを作る際の番地はリストと違い，$1200を加えた番地となることに注意してください。

また，コマンドとしてではなく，BASICのサブルーチンとして使うのであれば，$CFB8あたりからそのまま入力すれば，BASICから

```
CALL   &HCFB8
```

でハードコピーをとることができます。

その際，BASICに壊されないように，

```
CLEAR   &HCFB7   あるいは
LIMIT   &HCFB7
```

をプログラムの最初に実行します。

7. これで終りですが…

ジャングル(BASIC)の入口あたりでウロウロしている感じです。読者の中にはイライラしている人もいるかもしれませんが，そういう人は自分でズンズン踏み込んでください。

MZ-700／K／C／1200(SP-5030)

BASICの壁を越える

アプリケーション・プログラム DANGER GAME

杉浦 勇一

現在，マイコンの高級言語というと，「BASIC」がもっとも中心的な地位を占めています。

俗に言う〝BASIC全盛期〟が依然として続いているわけです。

BASICがこれだけ普及したのにはそれなりの理由があります。

たとえば，マシン語やアセンブラよりも人間の言葉に近いためプログラムが作りやすいとか，デバッグ（プログラムのミスを見つけて直すこと）がしやすい，などがあげられます。

そんなBASICにも越えられない「壁」とも言える限界があります。

その，もっとも致命的な限界の一つは，処理スピードの遅さです。

ゲームのプログラム，特にリアル・タイムゲームを作ったことのある人なら，実感としてわかると思いますが，BASICでは複雑な処理をさせると途端にトロくなってしまうのです。

プログラムを作ったことのない人でもゲームをしていて，スピードの遅さにじれったく思った経験があるでしょう。

また，もう一つBASICの限界が考えられます。

たとえば，同じBASICとはいっても，MZのBASICにはない命令が他の機種のBASICに付いているという場合です。

これをBASICだけで補うことができないとしたら，これもまた限界だと言えます。

そこで今回は，マシン語を使って，BASICの限界という壁を突き破ろう！　という主旨なのです。

ここではマシン語をBASICのひとつのコマンドとして，またはひとつのサブルーチンとして使っています。ですから，マシン語が理解できなくとも，それを使えるようになることを第1の目標とします。

文中に何を言っているのかわからないことがあっても，軽く流しておいてください。使っているうちにおのずと意味が通じてくるものです。

なおアプリケーションプログラムとして「DANGER GAME」を載せました。MZ-80K/C/1200用です。**MZ-700で使う場合はBASIC SP-5030を使ってください。**

また，マシン語自体は**MZ-80B/2000**でも同じですので，簡単に応用できると思いますので，読みとばしのないように。

前置きが長くなりましたがいよいよ本論です。

画面の記憶と復帰

まずは，画面をコントロールするテクニックを2つ紹介します。

現在，モニタ・テレビに描かれている画面を記憶し，好きなときに瞬時に見ることができれば便利でしょう。

ちょうどN-BASICでのGET@，PUT@と少し似ている働きです。

画面の記帳，復帰のマシン語プログラムが表1です。

表1　画面の記憶

アドレス	マシン語	機能
7000	01 E8 03	BC←03E8
7003	11 00 80	DE←8000
7006	21 00 D0	HL←D000
7009	ED B0	ブロック転送
700B	C9	RETURN

画面の復帰

アドレス	マシン語	機能
7010	01 E8 03	BC←03E8
7013	11 00 D0	DE←D000
7016	21 00 80	HL←8000
7019	ED B0	ブロック転送
701B	C9	RETURN

ちらっとだけ見てください。何を書いてあるのかわからない？

では，概略を説明します。

このプログラムは画面に書かれているデータをメモリの8000番地以下にコピーして，画面を記憶しているのですが，記憶した画面を見たいときは，逆に8000番地以下から画面にコピーすればよいのです。

マシン語のEDB0がコピーのためのブロック転送命令です。

まずますわからなくなった？　それでも結構です。とにかく次へ進みましょう。

画面の塗りつぶし

画面を塗りつぶす。それも一瞬にしてできるとなれば，たったこれだけのことでもいろいろと使い道があります。

表2がそのマシン語プログラムです。

表2　画面の塗りつぶし

アドレス	マシン語	機能
7020	21 00 D0	HL←D000
7023	01 E7 D3	BC←D3E7
7026	36 43	(HL)←43
7028	7C	A←H
7029	B8	A－B
702A	20 03	0でない時
702C	7D	A←L
702D	B9	A－C
702E	C8	0の時RETURN
702F	23	HL＋1
7030	18 F4	GOTO7026

これを実行すると画面全体が一瞬にしてまっ白になります。

実際にどんな感じになるのか，早く見てみたいですネ。

それでは次にこれらを実行させてみましょう。

おっと，その前にプログラムを入れなければなりません。

入力方法

システム・ソフトはまだBASICしか持っていない方のために，マシン語入力には表

3のプログラムを使います。

このプログラムを使えば，マシン語を16進表示のままDATA文に書き込むだけで入力されます。

ただし，DATA文の最後に＊＊を書くことを忘れないように。これが終り(END)のしるしです。

このプログラムは，メモリの7000番地から入れたいので，6FFF番地にLIMITをかけます。

また，［　　］の所にはプログラムの先頭番地7000(16進)を10進に直した28672が入ります。

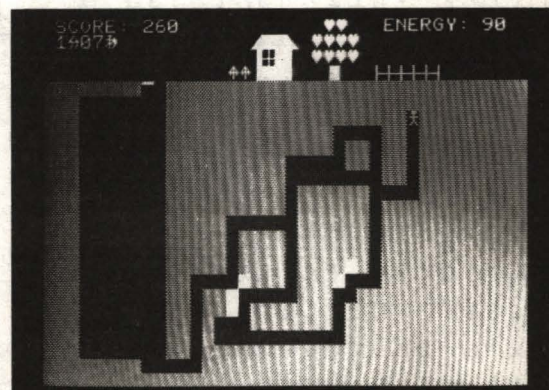
表3　マシン語入力プログラム

```
 10 LIMIT［プログラム先頭番地－1］
 20 FOR I=0 TO 999:D=0
 30 READ D$:IF D$="**" THEN 100
 40 A=ASC(D$)          -48:GOSUB 70
 50 A=ASC(RIGHT$(D$,1))-48:GOSUB 70
 60 POKE［プログラム先頭番地(10進)］+I,D:NEXT I
 70 IF A>16 THEN A=A-7
 80 IF(A>16)+(A<0) THEN PRINT"ERROR":END
 90 D=D*16+A:RETURN
  .      .
  .      .
  .      .
??? DATA ??,??,??
??? DATA **
```

DANGER GAME 行番号うちわけ

行　番　号	何をしているか
5	ゲームの説明サブへ
10 ～ 90	マシン語入力
100 ～ 130	初期設定
140 ～ 270	キー操作表示
280 ～ 400	ANALYSIS DATAの表示 410画面の記憶
430 ～ 670	地底地図の作成，表示 680画面の記憶
690 ～ 720	スタート処理 710画面の塗りつぶし
730 ～ 890	メイン（キー・スキャン，爆弾移動，時間表示）
900 ～ 990	爆弾，地雷の回収 判定
1100 ～ 1140	爆発処理 1000ゲーム画面の記憶 1130ゲーム画面に復帰
1150 ～ 1250	爆弾，地雷の回収処理(得点，人の移動)
1260 ～ 1590	ゲーム・オーバー表示 1400～1440地面のフラッシュ
1600 ～ 1690	回収成功処理（ボーナス計算）
1700 ～ 1840	ANALYSIS処理 1700ゲーム画面の記憶 1710 ANALYSIS画面表示 1840ゲーム画面の復帰
1850 ～ 1890	EARTH MAP処理 1850ゲーム画面の記憶 1860地底地図の表示 1880ゲーム画面の復帰
1900 ～ 1990	キー入力待ち
2000 ～ 2210	マシン語プログラムデータ
2220 ～ 2250	ANALYSISデータ
2260 ～	ゲームの説明サブ

変数表

変　数	何に使われているか
SC	スコア
HS	ハイ・スコア
BS	ボーナス
EN	エネルギー
K	爆弾，地雷の数
KX, KY	岩盤の座標
O	時間ぎれ判定
P	人の画面上の位置
C	人のキャラクタ
S	前方のキャラクタ
M	人の移動数
PA	人の地底地図上の位置
WX, WY	人の座標
T	KUF爆弾の地図上の位置
TC	KUF爆弾の移動先のキャラクタ
AN(4, 3)	ANALYSISデータ
DA(3)	表示用データ
AN	前方にある物体判定
R, RX, RY	雑用乱数
C1$, C2$	コメント

リスト1

```
10 LIMIT$6FFF
20 FOR I=0 TO 999:D=0
30 READ D$:IF D$="**" THEN 100
40 A=ASC(D$)           -48:GOSUB 70
50 A=ASC(RIGHT$(D$,1))-48:GOSUB 70
60 POKE 28672+I,D:NEXT I
70 IF A>16 THEN A=A-7
80 IF(A>16)+(A<0)THEN PRINT"ERROR":END
90 D=D*16+A:RETURN
100 PRINT"■";
110 GET A$
120 IF A$="1" THEN USR($7000):GOTO 180
130 IF A$="2" THEN USR($7010):GOTO 180
140 IF A$="3" THEN USR($7020):GOTO 180
150 PRINT"■■ ■";
160 PRINT A$;
170 GOTO 110
180 MUSIC"A0R3"
190 GOTO 110
200 REM■■ ｶﾞﾒﾝ ｷｵｸ ■■
210 DATA 01,E8,03
220 DATA 11,00,80
230 DATA 21,00,D0
240 DATA ED,B0
250 DATA C9
260 DATA 00,00,00,00
270 REM■■ ｶﾞﾒﾝ ﾌｯｷ ■■
280 DATA 01,E8,03
290 DATA 11,00,D0
300 DATA 21,00,80
310 DATA ED,B0
320 DATA C9
330 DATA 00,00,00,00
340 REM■■ ｶﾞﾒﾝ ﾇﾘﾂﾌﾞｼ ■■
350 DATA 21,00,D0
360 DATA 01,E7,D3
370 DATA 36,43
380 DATA 7C
390 DATA B8
400 DATA 20,03
410 DATA 7D
420 DATA B9
430 DATA C8
440 DATA 23
450 DATA 18,F4
460 DATA **
```

サンプルプログラム

リスト1のプログラムを入力してください。入力後，すぐにテープにSAVEしたほうがいいでしょう。

マシン語を使っていますので，暴走しないともかぎりません。できましたか？

RUNすると，しばらくしてカーソルが点滅します。

画面いっぱいにあなたの好きな絵でも描いてください。

そうしたら左上にある，①キーを押します。

これで今見えている画面が記憶されたわけです。

次に，③キーを押してください。

一瞬にして画面全体がまっ白になったでしょう。

これが「画面の塗りつぶし」です。

同じことをBASICで行なったとしたらこれほど速くはできません。

さて，先ほどの画面が本当に記憶されているかどうか確認しましょう。

②キーを押してください。どうですか？あなたが描いた画面が出てきたでしょう。

リスト2 **DANGER GAME**

```
5 GOSUB2260
10 LIMIT$6FFF
20 FOR I=0 TO 999:D=0
30 READ D$:IF D$="**" THEN 100
40 A=ASC(D$)           -48:GOSUB 70
50 A=ASC(RIGHT$(D$,1))-48:GOSUB 70
60 POKE 28672+I,D:NEXT I
70 IF A>16 THEN A=A-7
80 IF(A>16)+(A<0)THENPRINT"ERROR":END
90 D=D*16+A:RETURN
100 GOSUB1900:POKE$284B,0:DIMDA(3),AN(4,3):HS=0
110 FORI=0TO3:FORJ=0TO3
120 READD:AN(I,J)=D
130 NEXTJ,I
140 SC=0:K=11:PRINT"■"
150 PRINT,"*** KEY FUNCTION ***"
160 CURSOR3,6:PRINT"HI-SCORE";TAB(28);" TIME LIMIT"
170 PRINT"■    ";HS;TAB(31);"5ﾌﾝ ｲﾅｲ"
180 CURSOR3,13:PRINT"ANALYSIS";TAB(30);"EARTHMAP"
190 PRINT"      ┌─────┐";TAB(30);" ┌─────┐"
200 PRINT"      |SPACE|";TAB(30);"| C R |"
210 PRINT"      └─────┘";TAB(30);" └─────┘"
220 PRINT"  ENERGY -1";TAB(30);"ENERGY-20"
230 CURSOR19, 5:PRINT"┌─┐■■■|U|■■■└─┘■■|■■|■■■■|■■|"
240 CURSOR14,10:PRINT"┌─┐■■■|H|■■■└─┘■── 웃 ──"
250 CURSOR24,10:PRINT"┌─┐■■■|K|■■■└─┘"
260 CURSOR19,15:PRINT"┌─┐■■■|M|■■■└─┘"
270 GOSUB1900
280 PRINT"■"
290 CURSOR9,5:PRINT"*** ANALYSIS DATA ***"
300 PRINT"      ┌───────────────┬───┬───┬────┬────┐"
310 PRINT"      |               |C  |FE |ENG.|DEN.|"
320 PRINT"      ├───────────────┼───┼───┼────┼────┤"
330 PRINT"      |*:ﾌﾊﾂ ｼﾞﾗｲ     |0% |10%| 10U|5.0G|"
340 PRINT"      ├───────────────┼───┼───┼────┼────┤"
350 PRINT"      | :ﾌﾊﾂ ﾊﾞｸﾀﾞﾝ   |5% |15%| 10U|3.0G|":POKE53693,232
360 PRINT"      ├───────────────┼───┼───┼────┼────┤"
370 PRINT"      | :KUF ﾊﾞｸﾀﾞﾝ   |0% |15%|100U|5.5G|":POKE53773,232
380 PRINT"      ├───────────────┼───┼───┼────┼────┤"
390 PRINT"      |■:ｶﾞﾝﾊﾞﾝ       |5% |30%|  0U|8.5G|"
400 PRINT"      └───────────────┴───┴───┴────┴────┘"
410 USR($7000):GOSUB1900
```

これで「画面の塗りつぶし」，「画面の記憶・復帰」がどういうものかわかっていただけたと思います。

次に具体的な使い方について説明します。

以下をじっくりと読んでこれらのテクニックを自分のものにしてください。

記憶・復帰プログラムの説明

(1)○ ○ に画面の先頭番地を，16進法で左右を逆にして入れます。

(2)△ △ には先頭番地から数えた文字数を入れます。

ここでは画面全体を記憶するのですから，40字×25字＝1000字。

16進に直して03E8です。これを左右逆にして入れます。

(3)□ □ には画面のデータを記憶しておくところの先頭番地を入れます。

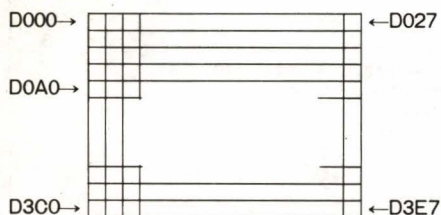
なお，記憶プログラムは7000番地より，復帰プログラムは7010番地より入力されていますので，BASICのUSR文でその番地を指定することで実行されます。

また，これらのプログラムはメモリのどこに置いてもそのままで使えます。

塗りつぶしプログラムの説明

(1)○ ○ に画面の先頭番地を，□ □ に画面の終りの番地を，16進で，左右を逆にして入れます。

ちなみにMZシリーズの画面の先頭番地はD000，終りの番地はD3E7です，(図1参照)。

図1　モニタ・画面の番地(16進法)

ここで，たとえば先頭番地をD0A0に変えて実行すると，画面の4行までは何もされず，5行目以下が塗りつぶされます。

サンプル・プログラムで試してみてください。　350行　DATA　21,00,D0を，

DATA　21,A0,D0

```
420 EN=100:S=0:BS=0:U=0:K=K+2
430 PRINT"■ SCORE:";SC;TAB(28);"ENERGY:";EN:PRINT"■■5900♣"
440 CURSOR0,4:FORI=1TO40:PRINT"‾";:NEXTI
450 PRINT"■";TAB(23);"♥♥"
460 PRINTTAB(17);"▲■■■ ♥♥♥♥"
470 PRINTTAB(17);"■H■ ♥♥♥♥"
480 PRINTTAB(15);"♣♣■■■■   ■   HHHHH
490 P=53377:C=202:POKEP,C
500 FORI=1TO5
510 KX=INT(RND(1)*37)+1
520 KY=INT(RND(1)*17)+6
530 FORJ=1TO10
540 RX=INT(RND(1)*3)-1
550 IF(KX+RX<0)+(KX+RX>38)THEN540
560 RY=INT(RND(1)*3)-1
570 IF(KY+RY<6)+(KY+RY>23)THEN560
580 KX=KX+RX:KY=KY+RY
590 CURSORKX,KY:PRINT"■";
600 NEXTJ,I
610 FORI=1TOK
620 R=INT(RND(1)*720)+53488:POKER,199
630 R=INT(RND(1)*720)+53488:POKER,232
640 NEXTI
650 T=INT(RND(1)*320)+53888
660 T=T-24576
670 TC=0
680 POKE$7005,112:USR($7000)
690 MUSIC"‾#B3‾B‾A‾G‾F‾E‾DCBAGFEDC‾B9"
700 CURSOR20,22:PRINT"START":GOSUB1900
710 POKE$7027,90:USR($7020)
720 TI$="000000":M=40:C=205:GOTO1190
730 GETA$
740 IFA$="U"THENM=-40:C=202:GOTO900
750 IFA$="H"THENM=-1 :C=203:GOTO900
760 IFA$="K"THENM= 1 :C=204:GOTO900
770 IFA$="M"THENM=40 :C=205:GOTO900
780 IFA$=" "THEN1700
790 IFA$="¥"THEN1850
800 IFVAL(TI$)>459THEN1330
810 IFVAL(TI$)<U+10THEN860
820 R=INT(RND(1)*3)-1:TR=T+R
830 R=INT(RND(1)*3)-1:TR=TR+R*40
840 IF(TR<28872)+(TR>29591)THEN860
850 U=VAL(TI$):POKET,TC:T=TR:TC=PEEK(T):POKET,232
860 CURSOR0,1:PRINT4-VAL(MID$(TI$,4,1))
870 CURSOR3,1
880 PRINTRIGHT$("0"+STR$(59-VAL(RIGHT$(TI$,2))),2)
890 GOTO730
900 IF(P+M<53408)+(P+M>54207)THEN730
910 POKEP,C
920 S=PEEK(P+M-24576)
930 POKEP+M,S:MUSIC"‾D0"
950 IFS=0  THEN1190
960 IFS=232THENON1-(M=40)GOTO1000,1150
970 IFS=199THENON1-(M=-40)GOTO1000,1170
980 MUSIC"_D2_#E"
990 GOTO730
1000 POKE$7005,120:USR($7000)
1010 W=P-53248:WY=INT(W/40):WX=W-WY*40
1020 FORI=3TO8:FORJ=1TO4
1030 RX=INT(RND(1)*I-I/2)+WX
1040 RY=INT(RND(1)*I-I/2)+WY
1050 IF(RX<0)+(RX>39)THENRX=WX
1060 IF(RY<5)+(RY>23)THENRY=WY
1070 CURSORRX,RY:PRINT"*";:POKEP+M,S
1080 MUSIC"A2":NEXTJ,I
1090 MUSIC"_D9"
1100 IFP+M=T+24576THEN1310
1110 EN=EN-10
1120 MUSIC"R9"
1130 POKE$7018,120:USR($7010)
1140 GOTO1190
1150 IFP+M=T+24576THEN1600
1160 SC=SC-40
1170 SC=SC+50:CURSOR7,0:PRINTSC
1180 POKEP+M,S:MUSIC"‾A2‾B‾CR5"
1190 POKEP,0
1200 POKEP-24576,0
1210 P=P+M
1220 CURSOR35,0:PRINTEN;"  "
1230 IFEN<1THEN1260
1240 POKEP,C
1250 GOTO800
```

(2) △には，画面を塗りつぶすときのキャラクタのディスプレイ・コードを16進で入れます。

このプログラムでは，〝■〟のディスプレイ・コード，43（16進）が入っています。

これも自分の好きなキャラクタに変えて試してみてください。

たとえば370行のDATA 36,43をDATA 36，53と変えて実行すれば画面全体が，ハートで埋めつくされます。

また，直接POKE文で書き換えてもかまいません。

この場合は POKE$7027，（ディスプレイコード）とします。

実行するときは，USR($7020) です。

さて，次にこれらのテクニックを応用したゲームプログラムを紹介します。

一つの応用例として参考にしてください。

DANGER GAME

このゲームはDANGER GAME(デンジャー・ゲーム) といい，まったくの頭脳ゲームです。

ストーリー

ここは昔の戦場で，地下のいたるところに不発爆弾や不発地雷が埋っています。

その不発爆弾のひとつにKUF爆弾というのがあり，それが何かのショックで作動しはじめました。

KUF爆弾は，地下を移動する能力を持っていて，ある時間が経過すると位置を変えてしまいます。

残された時間はあと5分。

大爆発を起す前に探し出してください。

ルール

KUF爆弾を探す手段としては，ANALYSIS(アナリシス) とEARTH·MAP（アース・マップ）があります。

アース・マップはその名のとおり，地底全体の地図を見せてくれますが，エネルギー消費が大きいのでやたらと使えません。それに，不発爆弾もKUF爆弾も外見はまったく同じなのでアースマップだけでは見つけ出すことができません。

アナリシスは，前方5つ先までの情報を調べて表示します。

```
1260 CURSOR35,0:PRINT"  0 ":MUSIC"A2BCDEFG5R"
1270 POKET+24576,232
1280 CURSOR7,7:PRINT"*** NO ENERGY ***"
1290 CURSOR9,10:PRINT"ｴﾈﾙｷﾞｰ ｶﾞ ﾅｸﾅﾘﾏｼﾀ"
1300 C2$="ﾍﾀｸｿ ﾓｳﾔﾒﾛ !":GOTO1510
1310 C1$="*** GAME OVER ***"
1320 C2$="ﾓｳｽｺｼ ｼﾝﾁｮｳ ﾆ ﾔﾚ !!":GOTO1360
1330 PRINT"■■  0分00秒"
1340 C1$="*** TIME UP ***"
1350 C2$="ﾄﾛｲﾅｧ──!  ﾓｯﾄ ﾊﾔｸ !!"
1360 FORI=1TO5
1370 POKET+24576,0  :MUSIC"R3￣A2"
1380 POKET+24576,232:MUSIC"R5"
1390 NEXTI
1400 POKE$7027,91 :USR($7020):MUSIC"_C2_D_E_F_G_A_B"
1410 POKE$7027,109:USR($7020):MUSIC"CDEFGAB"
1420 POKE$7027,198:USR($7020):MUSIC"￣C￣D￣E￣F￣G￣A￣B"
1430 POKE$7027,97 :USR($7020):MUSIC"￣C4C5_C6"
1440 POKE$7027,90 :USR($7020)
1450 POKET+24576,232
1460 CURSOR18,1:PRINT"+     ♥♥♥♥■■■♥♥"
1470 CURSOR19,2:PRINT"    ♥♥♥♥"
1480 CURSOR14,3:PRINT"\\■■■+   ◢♥   ///"
1490 CURSOR7,7:PRINTC1$
1500 CURSOR9,10:PRINT"KUFﾊﾞｸﾀﾞﾝ ｶﾞ ﾊﾞｸﾊﾂｼﾏｼﾀ"
1510 CURSOR9,12:PRINT"ﾖｯﾃ ﾎﾞｰﾅｽ ﾊ";BS;" ﾃﾝ"
1520 CURSOR9,14:PRINT"ｱﾅﾀ ﾉ SCORE ﾊ";SC;" ﾃﾞｽ"
1530 CURSOR9,16:IFSC>HSTHENHS=SC:PRINT"ｱﾅﾀ ｶﾞ HI-SCORE ﾃﾞｽ":GOTO1560
1540 PRINT"HI-SCORE ﾊ";HS;"ﾃﾞｽ"
1550 CURSOR9,18:PRINTC2$
1560 CURSOR24,21:PRINT"REPLAY ? (Y/N)"
1570 GETZ$:IFZ$="Y"THENON1-(BS=0)GOTO420,140
1580 IFZ$="N"THENCURSOR0,24:PRINT:END
1590 GOTO1570
1600 FORI=1TO7
1610 POKEP+M,0:MUSIC"R3￣A2"
1620 POKEP+M,232:MUSIC"R5"
1630 NEXTI
1640 MUSIC"C4DEFGAB￣C￣D￣E￣F￣4E￣D5￣C6BAG9"
1650 POKE$7027,90:USR($7020)
1660 CURSOR7,7:PRINT"*** CLEAR ! ***"
1670 CURSOR9,10:PRINT"KUF ﾊﾞｸﾀﾞﾝ ﾊ ﾌﾞｼﾞ ｶｲｼｭｳ ｻﾚﾏｼﾀ"
1680 C2$="ｱﾅﾀ ﾉ ﾄﾞﾘｮｸ ﾃﾞ ﾁｷｭｳ ﾊ ｽｸﾜﾚﾏｼﾀ"
1690 BS=(500-VAL(TI$))*2+EN*5:SC=SC+BS:GOTO1510
1700 POKE$7005,120:USR($7000)
1710 POKE$7018,116:USR($7010)
1720 EN=EN-1:PA=P-24576
1730 FORI=0TO3:DA(I)=0:NEXTI
1740 FORI=1TO5:PA=PA+M:AN=3
1750 IF PEEK(PA)=199THENAN=0
1760 IF PEEK(PA)=232THENAN=1:IFP+M*I=T+24576THENAN=2
1770 IF PEEK(PA)=0  THENAN=4
1780 FORJ=0TO3:DA(J)=DA(J)+INT(AN(AN,J)/I*100)/100
1790 NEXTJ,I
1800 CURSOR8,19:PRINT"C       FE       ENG.    DEN."
1810 PRINTTAB(7);DA(0);"%";TAB(14);DA(1);"%";
1820 PRINTTAB(21);DA(2);"U";TAB(28);DA(3);"G";
1830 MUSIC"A3R9":GOSUB1930:IFO=1THEN1330
1840 POKE$7018,120:USR($7010):GOTO1220
1850 POKE$7005,120:USR($7000)
1860 POKE$7018,112:USR($7010)
1870 POKEP,C:MUSIC"A3R9":GOSUB1930:IFO=1THEN1330
1880 POKE$7018,120:USR($7010)
1890 EN=EN-20:GOTO1220
1900 CURSOR26,22:PRINT"PUSH ANY KEY !":TEMPO7:MUSIC"￣A3"
1910 GETZ$:IFZ$=""THEN1910
1920 RETURN
1930 GETZ$:IFZ$=""THEN1950
1940 O=0:RETURN
1950 CURSOR0,1:PRINT4-VAL(MID$(TI$,4,1))
1960 CURSOR3,1
1970 PRINTRIGHT$("0"+STR$(59-VAL(RIGHT$(TI$,2))),2)
1980 IFVAL(TI$)>459THENO=1:RETURN
1990 GOTO1930
2000 DATA 01,48,03
2010 DATA 11,A0,74
2020 DATA 21,A0,D0
2030 DATA ED,B0
2040 DATA C9
2050 DATA 00,00,00,00
2060 DATA 01,48,03
2070 DATA 11,A0,D0
2080 DATA 21,A0,74
```

こちらは，エネルギー消費が少ないので何度も使えます。

この2つをうまく使い分けてください。

KUF爆弾の特徴として，ENG.が100であるのが目立ちます。

ですから，たとえばKUF爆弾が直前にある場合(図2)，アナリシスで調べると，〝ENG.100″と表示されます。

図2

図3

また，図3のように2歩前方にKUF爆弾がある場合，そのデータは$\frac{1}{2}$と小さくなります。

つまり，〝ENG.50″と表示されるわけです。

以下同様に，3歩前方なら$\frac{1}{3}$になります。

前方5つの間に物体が2つ以上あるときはそれらのデータの合計値が表示され，前方に何も何ければ，すべてのデータが0になります。

回収するとき，不発爆弾は上から，地雷は下から取り込まないと爆発します。

KUF爆弾も当然下から回収するわけです。その他細かいルールやキー操作は，プログラムを実行させると説明が出るので省略します。

地底のどこに地雷があり，どこに爆弾があるかを覚えている記憶力と，アナリシスを使って前方に何があるかを判断する力が必要なわけです。

内容が複雑そうですがそれほどでもありません。また難かしいほうが飽きないでしょう。

さて，今回のマシン語のテクニックや，DANGER GAMEはいかがだったでしょうか？

いろいろと応用して面白いプログラムを作ってください。

そして，良いプログラムができたら本誌に投稿しましょう。

```
2090 DATA ED,B0
2100 DATA C9
2110 DATA 00,00,00,00
2120 DATA 01,E7,D3
2130 DATA 21,A0,D0
2140 DATA 36,5A
2150 DATA 7C,B8
2160 DATA 20,03
2170 DATA 7D,B9
2180 DATA C8
2190 DATA 23
2200 DATA 18,F4
2210 DATA **
2220 DATA 0,10, 10,5
2230 DATA 5,15, 10,3
2240 DATA 0,15,100,5.5
2250 DATA 5,30,  0,8.5
2260 PRINT"■"
2270 PRINT"  ＿    ＿   ＿  ＿    ＿
2280 PRINT"／ ＼  ／＼ ／／／／  ／＼
2290 PRINT"■■＼ ■■／ ■＼■ ■■／
2300 PRINT"■■|■|■■|■■|■|■／
2310 PRINT"■|■|■／■|■■|■|＿
2320 PRINT"■|■|■|■|■■|■|／ ／
2330 PRINT"■|＿|■|■|■|■■|■■|
2340 PRINT"■／■／ ■|■|■|■|■／
2350 PRINT"■■／ ■／■／■／ ■■■
2360 PRINTTAB(24);" ＿＿   ＿＿ ＿
2370 PRINTTAB(24);"／   ／／  ＼ ／／
2380 PRINTTAB(24);"■■■／ ■■■／|■|
2390 PRINTTAB(24);"■|＿  ■|＿|■|
2400 PRINTTAB(24);"■／ ／ ■／ ■|■|
2410 PRINTTAB(24);"■■■／ ■■■／ ■／
2420 PRINTTAB(24);"■|＿＿ ■＼ ＼ ＿
2430 PRINTTAB(24);"■／  ／■■／／／
2440 PRINTTAB(24);"■■■／ ■／■／ ■／
2450 PRINTTAB(10);"<< デインヂャー ゲーム >>
2460 PRINT"■  ** STORY **
2470 PRINT"   ココ ハ ムカシ ノ センジョウ デ．チカ ノ イタルトコロ ニ
2480 PRINT"   フハツ バクダン( )ヤ．ジライ( ) ガ  ウマッテイル。■
2490 POKE53822,232
2500 POKE53832,199
2510 PRINT"   ナニカ ノ ショック デ  KUFバクダン ガ  サドウ シハ
2520 PRINT"   ジメタ。 アト 5分 デ  バクハツ スル。■
2530 PRINT"   アナタ ノ ニンム ハ KUFバクダン ヲ サガシダシテ
2540 PRINT"   カイシュウ スルコト デ  アル。 ケントウ ヲ イノル  !
2550 GOSUB1900
2560 PRINT"■
2570 PRINT"■  ** FUNCTION **■■
2580 PRINT"   ANALYSIS(アナリシス)...ブンセキ■
2590 PRINT"       C(タンソ ガンユウリョウ),FE(テツ ガンユウリョウ)
2600 PRINT"       ENG.(エネルギー),DEN.(ミツド) ヲ 5ツ サキ
2610 PRINT"       マデ  シラベテ ヒョウジ  スル。■
2620 PRINT"       エネルギー ショウヒ.....1 ユニット■■
2630 PRINT"   EARTH MAP(アース マップ)...チテイ チズ■
2640 PRINT"       チテイ ノ チズ  ヲ ヒョウジ  スル。■
2650 PRINT"       エネルギー ショウヒ....20 ユニット■■
2660 GOSUB1900
2670 PRINT"■
2680 PRINT"■    ** CHARACTER **■
2690 PRINT"       (フハツ ジライ)":POKE53413,199
2700 PRINT"         シタカラ カイシュウ シナイト バクハツ スル。
2710 PRINT"         カイシュウ ガ  セイコウ スルト...50 テン■
2720 PRINT"       (フハツ バクダン)":POKE53573,232
2730 PRINT"         ウエカラ カイシュウ シナイト バクハツ スル。
2740 PRINT"         カイシュウ ガ  セイコウ スルト...10 テン■
2750 PRINT"       (KUF バクダン)":POKE53733,232
2760 PRINT"         ウエカラ カイシュウ シナイト バクハツ スル。
2770 PRINT"         ミタトコロ フハツバクダン ト カワラナイ ガ
2780 PRINT"         チテイ ヲ 10秒 ニ イチド  イドウ スル。
2790 PRINT"         ソシテ 5分 ゴ  ニ バクハツ スル。
2800 PRINT"         カイシュウ ガ  セイコウ スルト...ボーナス■
2810 PRINT"      ■(ガンバン)
2820 PRINT"         コレ ガ  アルト ススムコト ガ  デキナイ。
2830 RETURN
2840 REM ---------------------------
2850 REM       ** DANGER GAME **
2860 REM          (SG-0001)
2870 REM       MZ-80K/C/1200/700
2880 REM         BASIC SP-5030
2890 REM          BY SUGIURA
2900 REM ---------------------------
```

MZ-700(S-BASIC) PROGRAMING KNOW HOW
画面の色を一瞬に変える マシン語チェンジ命令

きまこんクラブ 峰岸 順二

ゲームに使われるBASIC命令

あこがれのMZを手に入れ，さて，これからBASICを勉強しようと意欲に燃えているあなた——。マニュアルを開くといろいろなBASIC命令がぎっしりあるので，「こんなに勉強しなければならないなんて」と，とたんに不安になるのでは？

しかし，心配はいりません。S-BASICには104の命令がありますが，私たちがゲームに使うのは，ほんの15か20語で充分なのです。「5語で動かすパソコン」(注1) という入門書があるくらいですから。

ちなみに，パソコンゲームを何かひとつ選んで，どんな命令が使われているか調べてみましょう。例として，彩色四人麻雀，MZ-700版を取り上げました。これは，最初のパソコンとして52年にデビューしたTK-80BSを使っての私のオリジナル，四人麻雀ゲーム(注2) をMZ-700に移植したものです。機会があれば，いずれ発表したいと思っています。

結果を表1に示します。

麻雀のロジックは要するにIF～THENの塊であって151コも使っています。CURSOR, PRINT, COLORで画面を構成し，FOR～NEXT, GOTO, GOSUB, RETURNがプログラムの骨格となっています。

たった18語の命令と3種の関数の組み合わせにすぎませんが，これはどのゲームプログラムでも，この程度の基本語を使っているだけなのです。

これらの命令のひとつひとつの意味を覚えるのは数日でできてしまうと思います。この組み合わせでプログラムを作るわけですが，小さいプログラムを自分で作ってみること，数行でもいいから作ること，これが一番の近道です。

これはマシン語でも同じことです。

今月は，数語のマシン語命令を使って，どのゲームにも共通に使える，瞬間カラーチェンジプログラムに挑戦しましょう。

マシン語は難しいか 今月の勉強法

マシン語も数多くの命令がありますが，あれもこれも覚えようとすると，たちまち挫折してしまいます。

マシン語でも，プログラミングの要領はBASICとまったく同じですが，ただ命令が単純で，1を加えたり1を引いたり，取り扱える数が0～255までの数だったりの制限があります。

このように，単純な作業の積み重ねとなるので，プログラミングにはどうしても時間がかかります。しかしながら，そのスピードはすばらしく魅力的であり，一度マシン語の世界に踏み込むと，たちまちとりこになってしまうでしょう。

カラーチェンジ命令は，画面のカラーを一瞬で変える命令ですが，マシン語でないとできません。このプログラムを初めはBASICで組み，それをそのままマシン語に翻訳してみましょう。

ゲームプログラムのイントロ，途中，あるいは特にゲームオーバーの場面で使うと非常に効果があり，迫力が出ます。サブプログラムとして，あなたのゲーム作りに役に立つと思って取りあげました。

図1 テレビ画面のV-RAMアドレス

V-RAM

D000	D001	D002	…	D026	D027
D028	D029	D02A	…	D04E	D04F
…	…	…	…	…	…
D398	D399	D39A	…	D3BE	D3BF
D3C0	D3C1	D3C2	…	D3E6	D3E7

カラーV-RAM

D800	D801	D802	…	D826	D827
D828	D829	D82A	…	D84E	D84F
…	…	…	…	…	…
DB98	DB99	DB9A	…	DBBE	DBBF
DBC0	DBC1	DBC2	…	DBE6	DBE7

表1 四人麻雀ゲームで使われた命令数

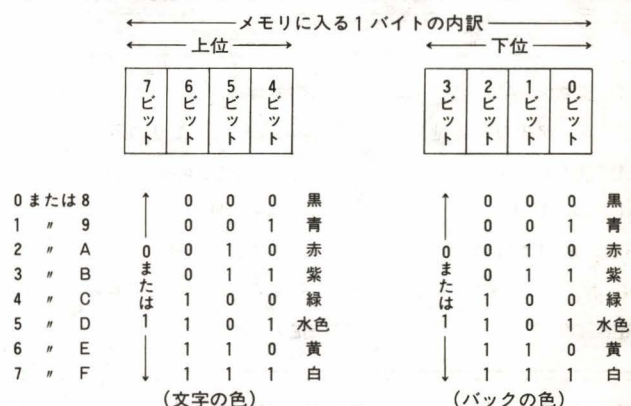
	命令	頻度		命令	頻度
1	IF～THEN	151	13	READ	5
2	PRINT	133	14	DATA	5
3	CURSOR	128	15	GET	2
4	GOSUB	82	16	DIM	1
5	REM	68	17	DEF	1
6	GOTO	64	18	ON～GOTO	1
7	FOR～NEXT	46	19	関数INT	19
8	RETURN	34	20	RND	18
9	COLOR	13	21	VAL	1
10	INPUT	12			
11	MUSIC	12		合計	802
12	TEMPO	6			

図2 D800～DBE7のメモリへ入る数とカラーの関係

メモリに入る1バイトの内訳

上位：7ビット 6ビット 5ビット 4ビット　　下位：3ビット 2ビット 1ビット 0ビット

	7ビット (0または1)	6ビット	5ビット	4ビット	文字の色	3ビット (0または1)	2ビット	1ビット	0ビット	バックの色
0または8		0	0	0	黒		0	0	0	黒
1 〃 9		0	0	1	青		0	0	1	青
2 〃 A		0	1	0	赤		0	1	0	赤
3 〃 B		0	1	1	紫		0	1	1	紫
4 〃 C		1	0	0	緑		1	0	0	緑
5 〃 D		1	0	1	水色		1	0	1	水色
6 〃 E		1	1	0	黄		1	1	0	黄
7 〃 F		1	1	1	白		1	1	1	白

（文字の色）　（バックの色）

画面の色を指定する原理

MZ-700のテレビ画面は40×25行，1000文字に区切られており，そのひとつひとつに一時記憶メモリ (V-RAM) が割り当てられています(マニュアル182ページ)。

そのアドレスはD000からD3E7　番地までです。このアドレスに00からFFまでのディスプレイコードが入ると，画面のその番地の場所にキャラクタ(マニュアル211ページ)が出るのです。このアドレスのバックおよび文字の色には，V-RAMと同じようなカラーV-RAMがあり，D800～DBE7がこのアドレスです。このメモリへ入る00～FFの256コの数によって，テレビの色が指定されるのです(図1)。このメモリへ入る数(16進で00～FF)とカラーの関係を図2にまとめました。

図に示すように，8ビットの数の右から6，5，4ビットで文字の色が，2，1，0ビットでバックの色が決まります。したがって，そのメモリに1を加えればバックの色が変わるし，00010001(16進で11H)を加えれば文字の色も変わります。

11Hを繰り返し加えていくと，7ビット目もやがては0から1に変わりますが，そうすると英大文字は小文字に，カナはひらがなになります。

テレビ画面の任意の1文字を変える

カラーを変えようとするテレビ画面のアドレス(D800～DBE7のうちのひとつ)に，1Hまたは11Hを加えることによって，上述のようにバック，またはバックと文字の色が変わります。これから使おうとするマシン語を7つに限定し，このマシン語に相当するBASIC命令を表2にまとめました。まずはこの表2だけのBASIC命令でプログラムが作れれば，これをマシン語に翻訳するのはやさしいものなのです。このLD(ロード)やINC，DEC，JPなどの命令を00～FFのマシン語コードに書き換え，ユーザーメモリへPOKEで書き込めばいいのです。この対応はマシン語入門書やインストラクション活用表を入手して見てください(注3)。

リスト1は，CURSOR 20，12の位置のバックの色を変えるものです。座標X，Yのアドレスは，

$D800＋X＋40＊Y

であることは容易にわかりますね。次々に変えたい場合，125および155行のFOR NEXT文を追加してください。画面のある範囲内の色を変える場合のために，リスト2を示します。

いよいよマシン語で

BASICでは多種の変数が使えますが，マシン語の世界では変数は限られています。レジスタと呼んでいますが，BASICと対応させる場合，これを変数と考えてください(表2注)。そして，この変数で扱える定数にも制限があって，A，B，C，D，E，H，Lでは00～FF(255)の数，HL，DEなどでは－32768から＋32767までの整数となっています。ただ，RAMメモリを定数の一時置き場として使えるので，まあ変数は制限されていてもどうにかやっていけるのです。

リスト3を見てください。オールBASICですが，行番号1000～1160のサブルーチンはマシン語に対応するBASICで書いてあります。まず，LIMIT　$BFFFとし，BASICプログラムが使うメモリを限定し，$C000以降をマシン語で使いましょう。定数や，カラーV-RAMのアドレスを一時的に記憶させておくRAMとして，$C100～$C105のアドレスを使います。

行番号160～220は$D9D2などのアドレスをD9とD2に分け，$C104，$C105に格納するための準備です。

カラーチェンジをBASICプログラムの中で使う場合，カラーV-RAMの計算や，X，Yなどの定数をメモリの中へPOKEするのはBASICで行ない，カラーチェンジだけをマシン語でするほうが容易です。リスト3をよく理解してください。

さて，いよいよマシン語で書きます。リスト4です。ただ，一部分でもマシン語をプログラムの中で使う場合，RUNの前に必ずSAVEしておいてください。1字の入れ違いから暴走してしまいますので。

00～FFの16進のマシン語コードをBASICに読み込ませるための命令が，　行番号1140～1190です。これは定石なので覚えてください。これによってマシン語が$C000から$C013までに書き込まれたのです。BYE [CR]，そしてDC000　C013をすれば，コードが書きこまれた状態がよく確認できます。

180行，USR($C000)は，C000からのマ

表2　カラーチェンジに使うマシン語～BASIC対応表

マシン語	BASIC
LD 変数[1]，A	変数＝A
LD A，(メモリの番地)	A＝PEEK(メモリの番地)
LD (HL)，A	POKE HL，A
ADD A，11H	A＝A＋11H
DEC B(またはC)	B＝B－1(C＝C－1)
JP NZ，アドレス	IF B<>0 THEN行番号
RET	RETURN

[1]変数　A，B，C，D，E，H，L，BC，DE，HLが使える

リスト1

```
10 REM *******************************
20 REM
30 REM      ﾘｽﾄ 1   1-POINT ｦ ｶｴﾙ
40 REM
50 REM *******************************
100 X=20:            REM X ｻﾞﾋｮｳ = 20
110 Y=12:            REM Y ｻﾞﾋｮｳ = 12
120 HL=$D800+X+40*Y:REM ｱﾄﾞﾚｽ ｹｲｻﾝ
130 A=PEEK(HL):      REM V-RAM ｼﾞｮｳﾎｳ
140 A=A+1:           REM V-RAM DATA+1
150 POKE HL,A:       REM DATAｦ V-RAM ﾍ
160 END
```

追加文

```
125 FOR I=1 TO 20

155 FOR J=1 TO 500:NEXT J:NEXT I
```

リスト2

```
10 REM ******************************
20 REM
30 REM     ﾘｽﾄ 2  ｱﾙ ﾊﾝｲ ﾉ ｲﾛｦ ｶｴﾙ
40 REM
50 REM ******************************
100 FOR X=15 TO 25: REM X ｦ 15 ｶﾗ 25
110 FOR Y=10 TO 15: REM Y ｦ 10 ｶﾗ 15
120 HL=$D800+X+40*Y:REM ｱﾄﾞﾚｽ ｹｲｻﾝ
130 A=PEEK(HL):     REM V-RAM ｼﾞｮｳﾎｳ
140 A=A+1:          REM V-RAM DATA+1
150 POKE HL,A:      REM DATAｦ V-RAM ﾍ
155 NEXT Y:NEXT X
160 END
```

シン語を実行せよ，という BASIC の命令でです。

1050行の PUSH 命令は，このマシン語サブプログラムへ入るまでの BASIC 命令を保護するためのもので，1120行 POP 命令は，ふたたびマシン語実行前の状態に復帰させるためのものです。

OUT(E3H), A および OUT(E1H), A は，始めと終わりのおまじないと今日のところは暗記してください(マニュアル181ページ)。

いよいよ，画面のある範囲の色を変えてみましょう。リスト5です。

画面の座標 (X, Y) から，横にP，タテにQの範囲の色を変えるプログラムで，理解しやすい内容にしたので，プログラムも長くなっています。主要変数，プログラムマップなどを図3にまとめました。

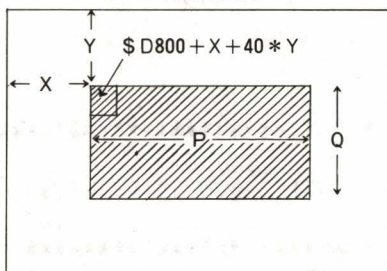
マシン語はC000番地からC035番地まで54バイトです。変数(レジスタ)BにPの値を入れ，1文字書くごとにB－1を行ない，Bが0になるまで繰り返します。0になったら1行下に移りますが，今，カラーチェンジしたアドレスに40を加え，Pを減じた数

HL＝HL＋40－P

のアドレスへ移ればいいわけです。この計算を1200～1210行で行なっています。

行番号1000以下のプログラムは，読みやすいように長く書いてありますが，機能だけを求めるのならば，リスト6に縮小することが可能です。

図3 カラーチェンジLIST-5の解説

カラーチェンジの画面範囲

プログラムマップ

行番号	内 容
100～150	X, Y, P, Qの値をメモリへ格納
160～230	座標(X, Y)のアドレスを求めHとLに分割しメモリへ格納
250～260	メインルーチン
1000～1260	マシン語のコードデータ
1270～1320	コードをC000～C035番地へ格納する

主要変数表

変 数	内 容	格 納 番 地
X	X座標の値	C100番地
Y	Y座標の値	C101番地
P	カラーチェンジの横幅	C102番地
Q	同タテの長さ	C103番地
AD$	色を変えるV-RAMアドレス	
H$	上のアドレス上位4ビット	
H	同上数値定数	C104番地
L$	V-RAM下位4ビット	
L	同上数値定数	C105番地

あとがき

このプログラムを作るのに，私もずいぶんエラーをしました。文字の脱落，感ちがい，；と：の違い，DATAの入れちがい，マシン語のコードの誤りなど山ほど。そしてリスト5では，大切な HL＝HL＋40＊Y の1200～1210行の命令を1180行の次に入れてしまって，予想もつかないカラーチェンジのあと暴走するエラーもしました。しかし，最後に笑えばいいのです。頑張ってください。今月のマシン語の命令を理解し，これが頭に残っていて，いつの日にかあなたのマシン語への入門のきっかけとなることを期待しています。

説明が長くなってしまって，今月はミニゲームを掲載することができませんでした。来月は一人マージャンプログラムとし，和了した瞬間にカラーチェンジするようにするのはどうでしょうか。また，もう一度，英文タイプの練習法について触れてみたいと考えています。

(注1) ダイヤモンドOA研究会　昭和57年　ダイヤモンド社発行

(注2) 峰岸　TK-80BS　4人麻雀ゲーム　I/O　54年12月号

(注３） (1)NEC μCOM-82インストラクション活用表
(2)Z80マイコンプログラムテクニック 電波新聞社

リスト３

```
10 REM ******************************
20 REM
30 REM    ﾘｽﾄ 3  ﾏｼﾝｺﾞ ｼｭﾐﾚｰｼｮﾝ
40 REM
50 REM ******************************
100 LIMIT $BFFF
110 X=20
120 Y=12
130 POKE $C100,X
140 POKE $C101,Y
150 AD=$D800+X+40*Y
160 AD$=HEX$(AD)       :REM ADｦ 16ｼﾝﾆ
170 H$=LEFT$(AD$,2)    :REM ｼﾞｮｳｲ 4BIT
180 H=VAL("$"+H$)      :REM ﾃｲｽｳﾆ ﾍﾝｶﾝ
190 L$=RIGHT$(AD$,2)   :REM ｶｲ 4 BIT
200 L=VAL("$"+L$)      :REM ﾃｲｽｳﾆ ﾍﾝｶﾝ
210 POKE $C104,H       :REM H ｦ $C104ﾆ

220 POKE $C105,L       :REM L ｦ $C105ﾆ
300 REM ****************************
310 REM
320 GOSUB 1000
330 REM
340 END
1000 REM ***************************
1010 REM
1020 REM        ﾏｼﾝｺﾞ ｻﾌﾞﾙｰﾁﾝ
1030 REM
1040 REM ﾚｼﾞｽﾀｰ ﾉ ﾀｲﾋ      :'PUSH AF...
1050 REM ﾒﾓﾘﾌﾞﾛｯｸ ｷﾘｶｴ     :'OUT (E3H),A
1060 A=PEEK($C104)        :'LD A,($C104)
1070 H=A                  :'LD H,A
1080 A=PEEK($C105)        :'LD A,($C103)
1090 L=A                  :'LD L,A
1100 GOSUB 1500
1110 A=PEEK(HL)           :'LD A,(HL)
1120 A=A+1                :'ADD A,01H
1130 POKE (HL),A          :'LD (HL),A
1140 REM ﾒﾓﾘﾌﾞﾛｯｸ ｷﾘｶｴ     :'OUT (E1H),A
1150 REM ﾚｼﾞｽﾀｰ ﾓﾄﾞｼ       :'POP AF...
1160 RETURN               :'RET
1500 REM ***************************
1510 REM
1520 REM  H ﾄ L ｦ ｸﾜｴﾃ HL ｦ ｹｲｻﾝ ｽﾙ
1530 REM
1540 H$=HEX$(H)
1550 L$=HEX$(L)
1560 IF L$="0" THEN L$="00"
1570 HL$=H$+L$
1580 HL=VAL("$"+HL$)
1590 RETURN
```

リスト４

```
100 REM ****************************
110 REM
120 REM   LIST-4 ﾏｼﾝｺﾞ ﾆﾖﾙ 1-POINT
130 REM          ﾉ ｶﾗｰ ﾁｪﾝｼﾞ
140 REM
150 REM ****************************
160 LIMIT $BFFF
170 GOSUB 1000
180 USR($C000)              :REM ﾒｲﾝﾙｰﾁﾝ
190 END
1000 REM ***************************
1010 REM
1020 REM   ﾏｼﾝｺﾞ ﾉ ﾖﾐｺﾐ ｻﾌﾞﾌﾟﾛｸﾞﾗﾑ
1030 REM
1040 REM ***************************
1050 DATA F5,C5,D5,E5     :'PUSH
1060 DATA D3,E3           :'OUT (E3H),A
1070 DATA 21,00,D9        :'LD HL,D900H
1080 DATA 7E              :'LD A,(HL)
1090 DATA C6,11           :'ADD 11H
1100 DATA 77              :'LD (HL),A
1110 DATA D3,E1           :'OUT (E1H),A
1120 DATA E1,D1,C1,F1     :'POP
1130 DATA C9              :'RET
1140 FOR I=$C000 TO $C013
1150 READ M$              :REM ﾖﾐｺﾐ
1160 M=VAL("$"+M$)        :REM ﾍﾝｶﾝ
1170 POKE I,M             :REM RAM ｶｷｺﾐ
1180 NEXT I
1190 RETURN
```

リスト５

```
10 REM ****************************
20 REM
30 REM     LIST-5 ｶﾗｰ ﾁｪﾝｼﾞ
40 REM
50 REM ****************************
100 LIMIT $BFFF
110 X=5:Y=3:P=30:Q=19
120 POKE $C100,X
130 POKE $C101,Y
140 POKE $C102,P
150 POKE $C103,Q
160 HL=$D800+X+40*Y
170 AD$=HEX$(HL)
180 H$=LEFT$(AD$,2)
190 H=VAL("$"+H$)
200 L$=RIGHT$(AD$,2)
210 L=VAL("$"+L$)
220 POKE $C104,H
230 POKE $C105,L
240 REM
250 GOSUB 1010
```

```
260 USR($C000)
270 REM
280 END
1000 REM ****************************
1010 REM   ﾏｼﾝｺﾞ ﾉ ﾖﾐｺﾐ ｻﾌﾞﾌﾟﾛｸﾞﾗﾑ
1020 REM ****************************
1030 REM
1040 DATA F5,C5,D5,E5 :'PUSH
1050 DATA D3,E3        :'OUT (E3H),A
1060 DATA 3A,04,C1     :'LD A,(C104H)
1070 DATA 67           :'LD H,A
1080 DATA 3A,05,C1     :'LD A,(C105H)
1090 DATA 6F           :'LD L,A
1100 DATA 3A,03,C1     :'LD A,(C103H)
1110 DATA 4F           :'LD C,A
1120 DATA 3A,02,C1     :'LD A,(C102H)
1130 DATA 47           :'LD B,A
1140 DATA 7E           :'LD A,(HL)
1150 DATA C6,11        :'ADD 11H
1160 DATA 77           :'LD (HL),A
1170 DATA 23           :'INC HL
1180 DATA 05           :'DEC B
1190 DATA C2,16,C0     :'JP NZ,C016H
1200 DATA 3A,02,C1,5F,3E,28:'HL=
1210 DATA 9B,11,00,00,5F,19:'    HL+40-P
1220 DATA 0D           :'DEC C
1230 DATA C2,12,C0     :'JP NZ,C012H
1240 DATA D3,E1        :'OUT (E1),A
1250 DATA E1,D1,C1,F1  :'POP
1260 DATA C9           :'RET
1270 FOR I=$C000 TO $C035
1280 READ M$
1290 M=VAL("$"+M$)
1300 POKE I,M
1310 NEXT I
1320 RETURN
```

リスト6

```
1000 REM ****************************
1010 REM
1020 REM   ﾏｼﾝｺﾞ ﾉ ﾖﾐｺﾐ ｻﾌﾞﾌﾟﾛｸﾞﾗﾑ
1030 REM      ｷﾏｺﾝ ｸﾗﾌﾞ     58/04/25
1040 REM      ｸﾗﾌﾞ FORESIGHT
1050 REM      ｸﾗｼｷ ﾏｲｺﾝ ｸﾗﾌﾞ ﾐﾈｷﾞｼ
1060 REM
1070 REM ****************************
1080 DATA F5,C5,D5,E5,D3,E3,3A,04,C1,67
1090 DATA 3A,05,C1,6F,3A,03,C1,4F,3A,02
1100 DATA C1,47,7E,C6,11,77,23,05,C2,16
1110 DATA C0,3A,02,C1,5F,3E,28,9B,11,00
1120 DATA 00,5F,19,0D,C2,12,C0,D3,E1,E1
1130 DATA D1,C1,F1,C9
1140 FOR I=$C000 TO $C035:READ M$
1150 M=VAL("$"+M$):POKE I,M:NEXT I
1160 RETURN
```

MZ-80K/C/1200/700
LIFE GAME

早稲田大学オーディオ研究会　河合 利彦

ジリリ〜ンと電話のベル。ふっとクラ〜イ影が脳裏をよぎる。こんな時間に電話をかけてくるのはYさんかKさんしかいない，と思って出たら，やっぱりYさん。明日はNAOKOちゃんとデートなのに——。

「あのねェ，先月がマシン語の入門って感じだったから，今月は続編という感じにしたいんだけど，なにか，マシン語のショートプログラムは入らないかなあ？」

エーッと叫んでみても仕方がないのです。こんなときのYさんの押しの強さには定評があるのですから……。

「マシン語ですか〜（明日はNAOKOちゃんと……）」

結局，来週までに1本完成させることを承諾させられてしまったわけです。

とはいっても時間がありません。昔々の本からネタをSearchしていたら，懐しいあのころ，ふと思いついたのがLIFE GAMEです。もまや知らない人も結構多いと思いますので，少々説明しておきましょう。

LIFE GAMEというのは，マイコンの創世紀に流行した遊びで，生物の増殖，死滅を極度にモデル化してシミュレートするものです。

生命体の最初の配置は人間が与えますが，その後は，愛機のMZまかせ。私達は，ボーッと見ているだけですが，その変化は千差万別，延々と世代を重ねたのちに，もとのパターンに戻るもの，いくらでも増殖していくもの，平行移動したり，こったものになると，大きな周期の繰り返しの中で，短い周期でパターンが振動するもの等々。

ただ見ているだけとは思えないほど楽しいものです。

というわけで，この「ゲーム」は，現在の意味でのゲームとはまったく異なり，純粋なシミュレーションです（ゲームだゲームだと喜んだ人は，ゴメンなさい）。

では，ルールの説明に移りましょう。

①盤面上のひとつの点に注目します。

②次に，その点のまわりの8個の点にある生命体の数を数えます。

③注目点が生命体のとき，まわりの生命体の数が2個か3個なら，注目点の生命体は生き残ります。1個以下や4個以上だと，注目点の生命体は死滅してしまいます。

③注目点が空白のとき，まわりの生命体の数が3個なら，そこに新しく生命体が誕生します。

以上のルールにそって，盤面全体を一度に動かす，という作業を繰り返していきます。少しややこしいですが，実際にプログラムを入れて，納得してください。

遊び方

プログラムを入れたら，念のために一応セーブしておきましょう。スタートアドレスは，C000Hです。

モニタからJC000 [CR] とすると，画面がクリアされます。このままでパターン設定をしても良いのですが，画面の範囲がわかりにくいという人は，[SHIFT] + [BREAK]

SAMPLE　No. 1

SAMPLE　No. 2

SANPLE　No. 3 “Queen-Bee”

をたたいてください。

生命体は"●",オダンゴです。その他のキャラクタは壁として扱われます。MZのスクリーンエディト機能をフルに使ってセットしてください。セットが終了したら,CRでスタートします。

いくつか,サンプルを書いておきますので参考にしてください。なお,写真のパターンは,撮影用に何回も止めたので,右上の,世代表示(GEN)は,いいかげんです。

この他にも,いろいろ面白いパターンが見つかると思います。

これは!　と思うようなものがあったら,「Oh!MZ LIFE GAME係」までお知らせください。

```
** OH!MZ CHECK SUM SYSTEM V1.0 **

ADR. +0 +1 +2 +3 +4 +5 +6 +7 :SUM.
C000 31 00 C0 D3 E4 3E C6 CD :4.79
C008 DC 0D 11 00 00 CD 03 00 :1.CA
C010 CD E0 C0 CD 68 C1 CD 1E :5.4E
C018 00 28 EF 21 00 D0 11 00 :2.19
C020 CC 01 FF 03 ED B0 16 01 :3.83
C028 1E 01 01 00 D0 CD 72 C0 :2.EF
C030 47 E6 20 20 16 78 FE 03 :2.FC
C038 28 2D FE 12 28 0D FE 13 :2.AB
-----------------------------------19C3
C040 28 09 01 00 CC CD CC C0 :3.57
C048 3E 00 77 1C 7B FE 27 20 :2.91
C050 D9 14 7A FE 18 20 D1 21 :3.8F
C058 00 CC 11 00 D0 01 FF 03 :2.B0
C060 ED B0 CD 8C C1 18 AF 01 :4.7F
C068 00 CC CD CC C0 3E 47 77 :4.21
C070 18 D9 3E 00 32 00 C2 E5 :3.08
C078 D5 C5 01 00 D0 CD CC C0 :4.C4
-----------------------------------1C93
C080 7E FE 00 28 10 FE 47 28 :3.21
C088 07 3E 20 32 00 C2 18 05 :1.76
C090 3E 10 32 00 C2 15 1D 01 :1.75
C098 00 D0 CD CC C0 06 03 0E :3.40
C0A0 03 78 FE 02 20 08 79 FE :3.1A
C0A8 02 20 03 C3 BA C0 7E FE :3.DE
C0B0 47 20 07 3A 00 C2 3C 32 :1.D8
C0B8 00 C2 23 0D 20 E3 11 25 :2.2B
-----------------------------------1447
C0C0 00 19 05 20 DA 3A 00 C2 :2.14
C0C8 C1 D1 E1 C9 D5 C5 6A 26 :5.66
C0D0 00 29 29 29 44 4D 29 29 :1.5E
C0D8 09 16 00 19 C1 09 D1 C9 :2.9C
C0E0 11 00 00 01 00 DQ CD CC :2.7B
C0E8 C0 06 28 3E FF 77 23 05 :2.CA
C0F0 20 FB 11 00 18 01 00 D0 :2.15
C0F8 CD CC C0 06 28 3E FF 77 :4.3B
-----------------------------------1709
C100 23 05 20 FB 01 00 D0 16 :2.2A
C108 17 1E 00 CD CC C0 3E FF :3.CB
C110 77 1E 27 CD CC C0 3E FF :4.52
C118 77 15 20 ED 21 47 C1 11 :2.D3
C120 0E D0 01 0B 00 ED B0 21 :2.A8
C128 52 C1 11 CB D3 01 11 00 :2.D4
C130 ED B0 21 63 C1 11 1C D0 :3.DF
C138 01 05 00 ED B0 21 00 00 :1.C4
-----------------------------------1839
C140 22 01 C2 CD 8C C1 C9 F0 :4.B8
C148 0C 09 06 05 F0 07 01 0D :1.25
C150 05 F0 F0 02 19 F0 0E 01 :2.FF
C158 0F 0B 0F 53 F0 66 F0 14 :2.D6
C160 2E 0B F0 07 05 0E 2E 4F :1.C0
C168 D3 E0 3E 01 32 00 00 3A :2.5E
C170 00 00 D3 E4 3D C0 3E 50 :3.42
C178 32 00 D8 21 00 D8 11 01 :2.15
-----------------------------------1527
C180 D8 01 FE 03 ED B0 3E 30 :3.E5
C188 32 D4 DB C9 FD 21 25 D0 :4.BD
C190 2A 01 C2 23 22 01 C2 DD :2.D2
C198 21 00 00 D9 01 10 05 D9 :1.E9
C1A0 06 00 DD 29 29 CB 10 78 :2.88
C1A8 D6 0A 38 03 47 DD 23 D9 :3.3B
C1B0 0D D9 20 EE 78 C6 20 FD :4.4F
C1B8 77 00 FD 2B DD E5 E1 DD :5.1F
-----------------------------------1C8E
C1C0 21 00 00 D9 0E 10 05 D9 :1.F6
C1C8 20 D6 C9 FF 00 00 FF FF :4.BC
C1D0 00 00 FF FF 00 00 FF FF :3.FC
C1D8 00 00 FF FF 00 00 FF FF :3.FC
C1E0 00 00 FF FF 00 00 FF FF :3.FC
C1E8 00 00 FF FF 00 00 FF FF :3.FC
C1F0 00 00 FF FF 00 00 FF FF :3.FC
C1F8 00 00 FF FF 00 00 FF FF :3.FC
-----------------------------------1E9A
```

```
   PAGE    1
 E STNR ADRS OBJECT     SOURCE STATEMENTS

      1                 ;
      2                 ;      LIFE GAME PROGRAM
      3                 ;                VER 1.0
      4                 ;
      5                 ;         BY T.K & NAOKO
      6                 ;
      7 0DDC            ?DPCT   EQU  0DDCH
      8 0003            GETK    EQU  0003H
      9 001E            BRKEY   EQU  001EH
     10 D000            VRAM    EQU  0D000H
     11 CC00            BUFF    EQU  0CC00H
     12 C200            ?PLATE  EQU  0C200H
     13 D800            CRAM    EQU  0D800H
     14 D025            DISPT   EQU  0D025H
     15 C201            ?GEN    EQU  ?PLATE+1
     16 0000            SPACE   EQU  00H
     17 0047            PLATE   EQU  47H
     18 0050            COLOR   EQU  50H
     19 00FF            WALL    EQU  0FFH
     20                 ;
     21                 ; SET PATTERN
     22                 ;
     23 0000            ORG  0C000H
     24 C000 3100C0             LD   SP,0C000H
     25 C003 D3E4               OUT  (0E4H),A
     26 C005 3EC6               LD   A,0C6H
     27 C007 CDDC0D             CALL ?DPCT
     28 C00A 110000     PRGST:  LD   DE,0000H
     29 C00D CD0300             CALL GETK
     30 C010 CDE0C0             CALL SCRN
     31 C013 CD68C1             CALL CLC
     32                 ;
     33                 ; MAIN ROUTINE
     34                 ;
     35 C016 CD1E00     START:  CALL BRKEY
     36 C019 28EF               JR   Z,PRGST
     37 C01B 2100D0             LD   HL,VRAM
     38 C01E 1100CC             LD   DE,BUFF
     39 C021 01FF03             LD   BC,03FFH
     40 C024 EDB0               LDIR
     41 C026 1601               LD   D,01H
     42 C028 1E01       LOOP1:  LD   E,01H
     43 C02A 0100D0     LOOP:   LD   BC,VRAM
     44 C02D CD72C0             CALL SEEK
     45 C030 47                 LD   B,A
     46 C031 E620               AND  20H
     47 C033 2016               JR   NZ,MPSS
     48 C035 78                 LD   A,B
     49 C036 FE03               CP   03H
     50 C038 282D               JR   Z,SET
     51 C03A FE12               CP   12H
     52 C03C 280D               JR   Z,MPSS
     53 C03E FE13               CP   13H
     54 C040 2809               JR   Z,MPSS
     55 C042 0100CC             LD   BC,BUFF
     56 C045 CDCCC0             CALL CALC
     57 C048 3E00               LD   A,SPACE
     58 C04A 77                 LD   (HL),A
     59 C04B 1C         MPSS:   INC  E
     60 C04C 7B                 LD   A,E
```

```
   PAGE     2
E STNR ADRS OBJECT     SOURCE STATEMENTS

   61 C04D FE27               CP    39
   62 C04F 20D9               JR    NZ,LOOP
   63 C051 14                 INC   D
   64 C052 7A                 LD    A,D
   65 C053 FE18               CP    24
   66 C055 20D1               JR    NZ,LOOP1
   67 C057 2100CC             LD    HL,BUFF
   68 C05A 1100D0             LD    DE,VRAM
   69 C05D 01FF03             LD    BC,03FFH
   70 C060 EDB0               LDIR
   71 C062 CD8CC1             CALL  COUNT
   72 C065 18AF               JR    START
   73                  ;
   74                  ; SET PLATE
   75                  ;
   76 C067 0100CC      SET:   LD    BC,BUFF
   77 C06A CDCCC0             CALL  CALC
   78 C06D 3E47               LD    A,PLATE
   79 C06F 77                 LD    (HL),A
   80 C070 18D9               JR    MPSS
   81                  ;
   82                  ;SEEK SURROUNDING PLATES
   83                  ;
   84                  ;   X ... E
   85                  ;   Y ... D
   86                  ;   N ... ACC
   87                  ;
   88 C072 3E00        SEEK:  LD    A,00H
   89 C074 3200C2             LD    (?PLATE),A
   90 C077 E5                 PUSH  HL
   91 C078 D5                 PUSH  DE
   92 C079 C5                 PUSH  BC
   93 C07A 0100D0             LD    BC,VRAM
   94 C07D CDCCC0             CALL  CALC
   95 C080 7E                 LD    A,(HL)
   96 C081 FE00               CP    SPACE
   97 C083 2810               JR    Z,SPSS
   98 C085 FE47               CP    PLATE
   99 C087 2807               JR    Z,SPSS3
  100 C089 3E20               LD    A,20H
  101 C08B 3200C2             LD    (?PLATE),A
  102 C08E 1805               JR    SPSS
  103 C090 3E10        SPSS3: LD    A,10H
  104 C092 3200C2             LD    (?PLATE),A
  105 C095 15          SPSS:  DEC   I
  106 C096 1D                 DEC   E
  107 C097 0100D0             LD    BC,VRAM
  108 C09A CDCCC0             CALL  CALC
  109 C09D 0603               LD    B,3
  110 C09F 0E03        SLOOP1: LD   C,3
  111 C0A1 78          SLOOP: LD    A,B
  112 C0A2 FE02               CP    2
  113 C0A4 2008               JR    NZ,SPSS1
  114 C0A6 79                 LD    A,C
  115 C0A7 FE02               CP    2
  116 C0A9 2003               JR    NZ,SPSS1
  117 C0AB C3BAC0             JP    SPSS2
  118                  ;
  119                  ; FIND A PLATE
  120                  ;


   PAGE     3
E STNR ADRS OBJECT     SOURCE STATEMENTS

  121 C0AE 7E          SPSS1: LD    A,(HL)
  122 C0AF FE47               CP    PLATE
  123 C0B1 2007               JR    NZ,SPSS2
  124 C0B3 3A00C2             LD    A,(?PLATE)
  125 C0B6 3C                 INC   A
  126 C0B7 3200C2             LD    (?PLATE),A
  127                  ;
  128                  ; LOOP END
  129                  ;
  130 C0BA 23          SPSS2: INC   HL
  131 C0BB 0D                 DEC   C
  132 C0BC 20E3               JR    NZ,SLOOP
  133 C0BE 112500             LD    DE,37
  134 C0C1 19                 ADD   HL,DE
  135 C0C2 05                 DEC   B
  136 C0C3 20DA               JR    NZ,SLOOP1
  137 C0C5 3A00C2             LD    A,(?PLATE)
  138 C0C8 C1                 POP   BC
  139 C0C9 D1                 POP   DE
  140 C0CA E1                 POP   HL
  141 C0CB C9                 RET
  142                  ;
  143                  ; CALCULATE : HL=(D*40+E)+BC
  144                  ;
  145 C0CC D5          CALC:  PUSH  DE
  146 C0CD C5                 PUSH  BC
  147 C0CE 6A                 LD    L,D
  148 C0CF 2600               LD    H,00H
  149 C0D1 29                 ADD   HL,HL
  150 C0D2 29                 ADD   HL,HL
  151 C0D3 29                 ADD   HL,HL
  152 C0D4 44                 LD    B,H
  153 C0D5 4D                 LD    C,L
  154 C0D6 29                 ADD   HL,HL
  155 C0D7 29                 ADD   HL,HL
  156 C0D8 09                 ADD   HL,BC
  157 C0D9 1600               LD    D,00H
  158 C0DB 19                 ADD   HL,DE
  159 C0DC C1                 POP   BC
  160 C0DD 09                 ADD   HL,BC
  161 C0DE D1                 POP   DE
  162 C0DF C9                 RET
  163                  ;
  164                  ; INITIALIZE SCREEN
  165                  ;
  166 C0E0 110000      SCRN:  LD    DE,0000H
  167 C0E3 0100D0             LD    BC,VRAM
  168 C0E6 CDCCC0             CALL  CALC
  169 C0E9 0628               LD    B,40
  170 C0EB 3EFF               LD    A,WALL
  171 C0ED 77          ZLOOP: LD    (HL),A
  172 C0EE 23                 INC   HL
  173 C0EF 05                 DEC   B
  174 C0F0 20FB               JR    NZ,ZLOOP
  175 C0F2 110018             LD    DE,1800H
  176 C0F5 0100D0             LD    BC,VRAM
  177 C0F8 CDCCC0             CALL  CALC
  178 C0FB 0628               LD    B,40
  179 C0FD 3EFF               LD    A,WALL
  180 C0FF 77          ZLOOP1: LD   (HL),A


   PAGE     4
E STNR ADRS OBJECT     SOURCE STATEMENTS

  181 C100 23                 INC   HL
  182 C101 05                 DEC   B
  183 C102 20FB               JR    NZ,ZLOOP1
  184 C104 0100D0             LD    BC,VRAM
  185 C107 1617               LD    D,23
  186 C109 1E00        ZLOOP2: LD   E,00H
  187 C10B CDCCC0             CALL  CALC
  188 C10E 3EFF               LD    A,WALL
  189 C110 77                 LD    (HL),A
  190 C111 1E27               LD    E,39
  191 C113 CDCCC0             CALL  CALC
  192 C116 3EFF               LD    A,WALL
  193 C118 77                 LD    (HL),A
  194 C119 15                 DEC   D
  195 C11A 20ED               JR    NZ,ZLOOP2
  196 C11C 2147C1             LD    HL,LETTER
  197 C11F 110ED0             LD    DE,0D00EH
  198 C122 010B00             LD    BC,000BH
  199 C125 EDB0               LDIR
  200 C127 2152C1             LD    HL,LETTER1
  201 C12A 11CBD3             LD    DE,0D3CBH
  202 C12D 011100             LD    BC,0011H
  203 C130 EDB0               LDIR
  204 C132 2163C1             LD    HL,LETTER2
  205 C135 111CD0             LD    DE,0D01CH
  206 C138 010500             LD    BC,0005H
  207 C13B EDB0               LDIR
  208 C13D 210000             LD    HL,0000H
  209 C140 2201C2             LD    (?GEN),HL
  210 C143 CD8CC1             CALL  COUNT
  211 C146 C9                 RET
  212 C147 F00C        LETTER: DEFB 0F0H,0CH
  213 C149 0906               DEFB  09H,06H
  214 C14B 05F0               DEFB  05H,0F0H
  215 C14D 0701               DEFB  07H,01H
  216 C14F 0D05               DEFB  0DH,05H
  217 C151 F0                 DEFB  0F0H
  218 C152 F002        LETTER1:DEFB 0F0H,02H
  219 C154 19F0               DEFB  19H,0F0H
  220 C156 0E01               DEFB  0EH,01H
  221 C158 0F0B               DEFB  0FH,0BH
  222 C15A 0F53               DEFB  0FH,53H
```

```
  223 C15C F066                DEFB 0F0H,66H
  224 C15E F014                DEFB 0F0H,14H
  225 C160 2E0B                DEFB 2EH,0BH
  226 C162 F0                  DEFB 0F0H
  227 C163 0705      LETTER2:DEFB 07H,05H
  228 C165 0E2E                DEFB 0EH,2EH
  229 C167 4F                  DEFB 4FH
  230                ;
  231                ; CLEAR COLOR RAM
  232                ;
  233 C168 D3E0      CLC:      OUT  (0E0H),A
  234 C16A 3E01                LD   A,01H
  235 C16C 320000              LD   (0000H),A
  236 C16F 3A0000              LD   A,(0000H)
  237 C172 D3E4                OUT  (0E4H),A
  238 C174 3D                  DEC  A
  239 C175 C0                  RET  NZ
  240 C176 3E50                LD   A,COLOR


   PAGE     5
E STNR ADRS OBJECT     SOURCE STATEMENTS

  241 C178 3200D8              LD   (CRAM),A
  242 C17B 2100D8              LD   HL,CRAM
  243 C17E 1101D8              LD   DE,CRAM+1
  244 C181 01FE03              LD   BC,03FEH
  245 C184 EDB0                LDIR
  246 C186 3E30                LD   A,30H
  247 C188 32D4DB              LD   (0DBD4H),A
  248 C18B C9                  RET
  249                ;
  250                ; DISPLAY GENERATION
  251                ;
  252 C18C FD2125D0  COUNT:    LD   IY,DISPT
  253 C190 2A01C2              LD   HL,(?GEN)
  254 C193 23                  INC  HL
  255 C194 2201C2              LD   (?GEN),HL
  256 C197 DD210000            LD   IX,0000H
  257 C19B D9                  EXX
  258 C19C 011005              LD   BC,0510H
  259 C19F D9                  EXX
  260 C1A0 0600      GLOOP1: LD   B,00H
  261 C1A2 DD29      GLOOP:    ADD  IX,IX
  262 C1A4 29                  ADD  HL,HL
  263 C1A5 CB10                RL   B
  264 C1A7 78                  LD   A,B
  265 C1A8 D60A                SUB  0AH
  266 C1AA 3803                JR   C,GPSS
  267 C1AC 47                  LD   B,A
  268 C1AD DD23                INC  IX
  269 C1AF D9        GPSS:     EXX
  270 C1B0 0D                  DEC  C
  271 C1B1 D9                  EXX
  272 C1B2 20EE                JR   NZ,GLOOP
  273 C1B4 78                  LD   A,B
  274 C1B5 C620                ADD  A,20H
  275 C1B7 FD7700              LD   (IY),A
  276 C1BA FD2B                DEC  IY
  277 C1BC DDE5                PUSH IX
  278 C1BE E1                  POP  HL
  279 C1BF DD210000            LD   IX,0000H
  280 C1C3 D9                  EXX
  281 C1C4 0E10                LD   C,10H
  282 C1C6 05                  DEC  B
  283 C1C7 D9                  EXX
  284 C1C8 20D6                JR   NZ,GLOOP1
  285 C1CA C9                  RET

ERRORS     0


SYMBOL  ADRS   SYMBOL  ADRS   SYMBOL  ADRS

?DPCT   0DDC   ?GEN    C201   ?PLATE  C200
BRKEY   001E   BUFF    CC00   CALC    C0CC
CLC     C168   COLOR   0050   COUNT   C18C
CRAM    D800   DISPT   D025   GETK    0003
GLOOP1  C1A0   GLOOP   C1A2   GPSS    C1AF
LETTER  C152   LETTER  C163   LETTER  C147
LOOP1   C028   LOOP    C02A   MPSS    C04B
PLATE   0047   PRGST   C00A   SCRN    C0E0
SEEK    C072   SET     C067   SLOOP1  C09F
SLOOP   C0A1   SPACE   0000   SPSS1   C0AE
SPSS2   C0BA   SPSS3   C090   SPSS    C095
START   C016   VRAM    D000   WALL    00FF
ZLOOP1  C0FF   ZLOOP2  C109   ZLOOP   C0ED
```

X1 (HuBASIC)

Z-80逆アセンブラと命令語の解析

イッティ・リッターポーン

Ｘ１ファンの皆さん，こんにちは。あなたのＸ１の調子はいかがですか？　Ｘ１の名もようやくマイコン界に拡がり，好評を得ているようです。われわれＸ１ユーザーにとってもうれしいことです。しかし，いくらハードの優れたＸ１でも，もっとたくさんのソフトがないと将来は暗いものになってしまいますね。ソフト作りに精を出しましょう。

というわけで，今回はＸ１用Z-80逆アセンブラを作りましたので，ご紹介します。

逆アセンブラとは何か

今日のようにマイコンがブームになってくると，逆アセンブラを知らないという人はあまりいないのかもしれませんが，一応かんたんに説明しておきます。

ご存知のとおり，マイコンにとっていちばんわかりやすい言葉はマシン語です。しかし，このマシン語，それを直接作ることはとてもたいへんで，効率が悪いものです。そこで，マシン語プログラムの作成を効率よくやるために，アセンブリ言語が作られました。アセンブリ言語はBASIC言語ほど人間にわかりやすくはないのですが，マシン語よりはよほどわかりやすく，一言でいうと，マシン語の命令を人間にわかりやすい言葉で記述した言語なのです。そして，アセンブラとは，要するに，アセンブリ言語で作られたプログラムをマシン語に変換するプログラムです。

また逆に，マシン語のプログラムがあるとします。やはり，このマシン語プログラムを理解するのに，直接プログラムをたどっていくのは容易なことではありません。逆アセンブラとは，マシン語プログラムを人間にわかりやすいアセンブリ言語に翻訳してくれるプログラムです。したがって，どんな言語で作ってもちゃんと正しく翻訳してくれればいいわけです。

一般にどんなプログラムでもそうですが，オールマシン語で作ると，メモリをBASICで作るほどは，大きく必要としないので，都合がいいわけです。特に，逆アセンブラプログラムはなるべく小さいほうが，逆アセンブルしたいプログラムの邪魔にならずにいいのです。X1の場合，HuBASICのモニタ(MON)で，マシン語で逆アセンブラプログラムを作ってもいいのですが，何といってもHuBASIC自体も約40Kバイトなのであまり意味がありません。もちろん，BASIC抜きで，マシン語を使って逆アセンブラを作れる人はいいのですが，ぼくにはとてもできません。したがって，今回は，HuBASIC下で，BASICを使って逆アセンブラを作りました。X1, HuBASICの内部解析などに役立てばと思います。

プログラムの入力方法

このプログラムは，X1のテープモード(CZ-8CB01)でも，ディスクモード(CZ-8FB01)でも使えます。

まずHuBASICをロードして，次にリストを打ち込んでください。終わったらテープかディスケットにセーブしOKです。

プログラムリスト1

```
10 ' ********************************************
15 ' *                                          *
20 ' *      DISASSEMBLER (Z-80) for X1          *
25 ' *                                          *
30 ' *          by  Itti  Rittaporn             *
35 ' *                                          *
40 ' *             1983  April 29               *
45 ' *                                          *
50 ' *   (TAPE & DISK  COMMON VERSION )         *
55 ' *                                          *
60 ' ********************************************
65 '****  MAIN ROUTINE  ****
70 DEFINT B-Z:DIM A$(79),B$(3),C$(7),D$(7),E$(3):FORI=0 TO 79:READ A$(I):NEXT:FO
RI=0 TO 3:READ B$(I):NEXT:FOR I=0 TO 7:READ C$(I):NEXT:FOR I=0 TO 7:READ D$(I):N
EXT:LI=1:LJ=1
75 WIDTH80:CONSOLE0,25:GOSUB85:EF=0:ON GD GOSUB 125,210:COLOR3:PRINT:PRINT:PRINT
"to continue input..[1]";:I=1:INPUT" ",I:IF I=1 THEN 75 ELSE CLS4:END
80 '
85 CLS4:COLOR6,0:LINE (80,0)-(500,55),PSET,B:COLOR4,0:LOCATE23,1:PRINT"Z-80 DISA
SSEMBLER for X1":LOCATE25,3:PRINT"by Itti Rittaporn":LOCATE26,5:PRINT"1983 APRIL
 29"
90 '
95 PR=0:LOCATE28,9:PRINT"PRINTER  ONLINE...":COLOR2:LOCATE47,9:PRINT" 1":COLOR4:
LOCATE37,11:PRINT"OFFLINE..":COLOR2:LOCATE47,11:PRINT"[0]";:COLOR4:INPUT" ",PR:I
F PR<>1 AND PR <>0 THEN 95
100 LOCATE28,15:PRINT"start address =":COLOR2:LOCATE44,15:INPUT" ",I$:SA=VAL("&H
```

```
"+I$):IF SA<0 THEN SA=SA+65536!
105 COLOR4:LOCATE28,16:PRINT"end    address =":COLOR2:LOCATE44,16:INPUT" ",I$:EA=
VAL("&H"+I$):IF EA<0 THEN EA=EA+65536
110 IF EA<SA THEN 105
115 GD=1:COLOR4:LOCATE28,18:PRINT"Disassembler mode.. [1]":LOCATE28,19:PRINT"Data
          mode...2":COLOR2:LOCATE50,19:INPUT" ",GD:IF GD<>1 AND GD<>2 THEN 115 EL
SE RETURN
120 '
125 I=1:COLOR4:LOCATE28,21:PRINT"ONE ROW.. [1]":LOCATE28,22:PRINT"TWO ROWS..2":CO
LOR2:LOCATE40,22:INPUT" ",I:COLOR4:IF I<1 OR I>2THEN 125
130 CLS4:WIDTH80:ON I GOSUB 140,160:RETURN
135 '
140 GOSUB 280:PRINT G$:GOSUB 845:IF PR=1 THEN LPRINT G$
145 IF PR=1 THEN LI=LI+1:IF LI=63 THEN LI=1:PRINT:PRINT:LPRINT CHR$(12)
150 IF EF=0 THEN 140 ELSE RETURN
155 '
160 A1=SA:FOR I1=1 TO 62:GOSUB 280:NEXT:A2=SA:SA=A1:EF=0:IF A2>EA THEN GOSUB 140
 ELSE GOSUB 175
165 IF EF=0 THEN 160 ELSE RETURN
170 '
175 SA=A1:GOSUB 280:A1=SA:PRINT G$;:GOSUB 845:IF PR=1 THEN LPRINT G$;
180 IF A2>EA THEN PRINT:IF PR=1 THEN LPRINT:GOTO 190 ELSE 190
185 SA=A2:GOSUB 280:A2=SA:PRINT TAB(40);G$:IF PR=1 THEN LPRINT TAB(39);G$
190 IF PR=1 THEN LI=LI+1:IF LI<63 THEN 175 ELSE LI=1:SA=A2:PRINT:PRINT:LPRINT CH
R$(12):RETURN
195 LJ=LJ+1: IF LJ<63 THEN 175 ELSE LJ=1:SA=A2:PRINT:PRINT
200 RETURN
205 '
210 CLS4:WIDTH80
215 G$="":GOSUB 680:GOSUB 860:E$(0)=EF$:AD$=RIGHT$("000"+HEX$(SA-1),4)+" ":OP$="
DB    "+N$:IF SA>EA THEN EF=1 ELSE GOSUB 260
220 G$=LEFT$(AD$+G$+"          ",13)+"    "+OP$:COLOR5:PRINT G$;:GOSUB 875:GOSUB 845:
IF PR=0 THEN GOTO 245
225 IF (PR=1) AND (EF=1) THEN Y=BJ:GOTO 235
230 IF (PR=1) AND (EF=0) THEN Y=3 ELSE GOTO 245
235 G$=G$+"  ":FOR Z=0 TO Y:G$=G$+E$(Z):NEXT Z
240 LPRINT G$
245 IF PR=1 THEN LI=LI+1:IF LI=63 THEN LI=1:PRINT:PRINT:LPRINT CHR$(12)
250 IF EF=0 THEN 215 ELSE COLOR4:RETURN
255 '
260 BI=1
265 GOSUB 680:OP$=OP$+","+N$:GOSUB 860:E$(BI)=EF$:BI=BI+1:IF SA>EA THEN BJ=BI-1:
EF=1:BI=4
270 IF BI<4 THEN 265 ELSE RETURN
275 '
280 GOSUB 480:AD$=RIGHT$("000"+HEX$(SA),4)+" "+MC$:SA=SA+1:G$="":OP$="":GOSUB 51
5:G$=LEFT$(AD$+G$+"          ",13)+"    "+OP$:IF SA>EA THEN EF=1
285 RETURN
290 ' C1=1
295 ON C3 GOSUB 340,355,380,425,440,450,460,470:RETURN
300 ' C1=2
305 IF MC$="76" THEN OP$="HALT" ELSE OP$=A$(8)+C$(C2-1)+","+C$(C3-1)
310 RETURN
315 ' C1=3
320 OP$=A$(C2+22)+C$(C3-1):RETURN
325 ' C1=4
330 ON C3 GOSUB 520,530,545,555,585,595,620,630:RETURN
335 ' C1=1 & C3=1
340 OP$=A$(C2-1):IF C2>2 THEN GOSUB 660:OP$=OP$+E$
345 RETURN
350 ' C1=1 & C3=2
355 IF C2 AND 1 THEN 365:' C2=1,3,5,7 ---2650
360 OP$="ADD   HL,"+B$(C2/2-1):GOTO 370
365 GOSUB 495:OP$=A$(8)+B$(INT(C2/2))+","+NN$
370 RETURN
375 ' C1=1 & C3=3
```

```
380 OP$=A$(8)
385 IF C2<5 THEN OP$=OP$+A$(C2+18):GOTO 415
390 GOSUB 495:ON C2-4 GOTO 395,400,405,410
395 OP$=OP$+"("+NN$+"),HL":GOTO 415
400 OP$=OP$+"HL,("+NN$+")":GOTO 415
405 OP$=OP$+"("+NN$+"),A":GOTO 415
410 OP$=OP$+"A,("+NN$+")"
415 RETURN
420 ' C1=1 & C3=4
425 IF C2 AND 1 THEN OP$=A$(9)+B$(INT(C2/2)) ELSE OP$=A$(10)+B$(C2/2-1)
430 RETURN
435 ' C1=1 & C3=5
440 OP$=A$(9)+C$(C2-1):RETURN
445 ' C1=1 & C3=6
450 OP$=A$(10)+C$(C2-1):RETURN
455 ' C1=1 & C3=7
460 GOSUB 680:OP$=A$(8)+C$(C2-1)+","+N$:RETURN
465 ' C1=1 & C3=8
470 OP$=A$(C2+10):RETURN
475 '
480 MC=PEEK(SA):MC$=RIGHT$("0"+HEX$(MC),2)
485 RETURN
490 '
495 GOSUB 480:SA=SA+1:G$=G$+MC$:I=MC
500 GOSUB 480:SA=SA+1:G$=G$+MC$:NN$=RIGHT$("000"+HEX$(MC*256+I),4)+"H"
505 RETURN
510 '
515 C1=MC¥64+1:C2=((MC¥8) AND 7)+1:C3=(MC MOD 8)+1:ON C1 GOSUB 295,305,320,330:R
ETURN
520 OP$="RET  "+D$(C2-1):RETURN
525 ' C1=4 & C3=2
530 IF C2 AND 1 THEN IF C2=7 THEN OP$="POP  AF" ELSE OP$="POP  "+B$(INT(C2/2)) E
LSE OP$=A$(INT(C2/2)+30)
535 RETURN
540 ' C1=4 & C3=3
545 GOSUB 495:OP$="JP   "+D$(C2-1)+","+NN$:RETURN
550 ' C1=4 & C3=4
555 IF C2>4 THEN OP$=A$(C2+30):GOTO 575
560 IF C2=1 THEN GOSUB 495:OP$="JP   "+NN$:GOTO 575
565 IF C2=2 THEN GOSUB 640:GOTO 575
570 GOSUB 680:IF C2=3 THEN OP$="OUT  "+"("+N$+")"+",A" ELSE OP$="IN   A,"+"("+N$
+")"
575 RETURN
580 ' C1=4 & C3=5
585 GOSUB 495:OP$="CALL "+D$(C2-1)+","+NN$:RETURN
590 ' C1=4 & C3=6
595 IF C2 AND 1 THEN IF C2=7 THEN OP$="PUSH AF":GOTO 610 ELSE OP$="PUSH "+B$(INT
(C2/2)):GOTO 610
600 IF C2=2 THEN GOSUB 495:OP$="CALL "+NN$:GOTO 610
605 ON C2/2-1 GOSUB 740,750,810
610 RETURN
615 ' C1=4 & C3=7
620 GOSUB 680:OP$=A$(C2+38)+N$:RETURN
625 ' C1=4 & C3=8
630 OP$="RST  "+RIGHT$("0"+HEX$((C2-1)*8),2)+"H":RETURN
635 '
640 GOSUB 680:C1=MC¥64+1:C2=((MC¥8) AND 7):C3=(MC MOD 8):IF C1>1 THEN OP$=A$(C1+
45)+"  "+CHR$(48+C2)+","+C$(C3):GOTO 650
645 IF C2=6 THEN OP$="" ELSE OP$=A$(C2+50)+C$(C3)
650 RETURN
655
660 GOSUB 480:SA=SA+1:G$=MC$:IF MC> 127 THEN MC=MC-25
665 E$=RIGHT$("000"+HEX$(SA+MC),4)+"H"
670 RETURN
675 '
680 GOSUB 480:SA=SA+1:G$=G$+MC$:N$=MC$+"H"
685 RETURN
```

```
690 '
695 GOSUB 680:IF MC$="DD" OR MC$="ED" OR MC$="FD" THEN 730
700 IF MC$="29" THEN OP$="ADD   "+XY$+","+XY$:GOTO 730
705 IF MC$="36" THEN GOSUB 680:OP$="LD    ("+XY$+"+"+N$+"),":GOSUB 680:OP$=OP$+N$
:GOTO 730
710 IF MC$="E9" THEN OP$="JP    ("+XY$+")":GOTO 730
715 IF MC$="CB" THEN GOSUB 680:D$=N$:GOSUB 640:I=INSTR(OP$,"(HL"):IF I=0 THEN OP
$="":GOTO 730 ELSE OP$=LEFT$(OP$,I)+XY$+"+"+D$+")":GOTO 730
720 A0=SA:SA=SA-1:GOSUB 480:SA=SA+1:GOSUB 515:I=INSTR(OP$,"(HL)"):IF I>0 THEN GO
SUB 680:OP$=LEFT$(OP$,I)+XY$+"+"+N$+MID$(OP$,I+3):GOTO 730
725 I=INSTR(OP$,"HL"):IF I>0 THEN OP$=LEFT$(OP$,I-1)+XY$+MID$(OP$,I+2) ELSE SA=A
0:G$=LEFT$(G$,2):OP$=""
730 RETURN
735 '
740 XY$="IX":GOSUB 695:RETURN
745 '
750 GOSUB 680:C1=MC¥64:C2=((MC¥8) AND 7):C3=(MC MOD 8):IF C1=0 OR C1=3 THEN 800
ELSE IF C1=1 THEN ON C3+1 GOTO 755,760,765,770,775,780,785,790 ELSE 795
755 IF C2=6 THEN 800 ELSE OP$="IN    "+C$(C2)+",(C)":GOTO 800
760 IF C2=6 THEN 800 ELSE OP$="OUT   (C),"+C$(C2):GOTO 800
765 OP$="  HL,"+B$(INT(C2/2)):IF C2 AND 1 THEN OP$="ADC"+OP$:GOTO 800 ELSE OP$="
SBC"+OP$:GOTO 800
770 IF C2=4 OR C2=5 THEN 800 ELSE GOSUB 495:IF C2 AND 1 THEN OP$="LD    "+B$(INT(
C2/2))+",("+NN$+")":GOTO 800 ELSE OP$="LD    ("+NN$+"),"+B$(C2/2):GOTO 800
775 IF MC$="44" THEN OP$="NEG":GOTO 800 ELSE 800
780 IF C2=0 THEN OP$="RETN":GOTO 800 ELSE IF C2=1 THEN OP$="RETI":GOTO 800 ELSE
800
785 IF MC$="46" THEN OP$="IM    0":GOTO 800 ELSE IF MC$="56" THEN OP$="IM    1":GO
TO 800 ELSE IF MC$="5E" THEN OP$="IM    2":GOTO 800 ELSE 800
790 IF C2>5 THEN 800 ELSE OP$=A$(C2+74):GOTO 800
795 IF C2>3 OR C3<4 THEN OP$=A$((C2-4)*4+C3+58)
800 RETURN
805 '
810 XY$="IY":GOSUB 695:RETURN
815 '
820 DATA NOP,"EX    AF,AF'","DJNZ  ","JR    ","JR    NZ,","JR    Z,","JR    NC,","JR
 C,","LD    ","INC   ","DEC   ",RLCA,RRCA,RLA,RRA,DAA,CPL,SCF,CCF,"(BC),A","A,(BC)"
,"(DE),A","A,(DE)","ADD   A,","ADC   A,","SUB   ","SBC   A,","AND   ","XOR   ","OR    "
,"CP    "
825 DATA RET,EXX,JP    (HL),"LD    SP,HL","EX    (SP),HL","EX    DE,HL",DI,EI,"ADD
A,","ADC   A,","SUB   ","SBC   A,","AND   ","XOR   ","OR    ","CP    ",BIT,RES,SET,"RLC
  ","RRC   ","RL    ","RR    ","SLA   ","SRA   ",,"SRL   "
830 DATA LDI,CPI,INI,OUTI,LDD,CPD,IND,OUTD,LDIR,CPIR,INIR,OTIR,LDDR,CPDR,INDR,OT
DR,"LD    I,A","LD    R,A","LD    A,I","LD    A,R",RRD,RLD
835 '
840 DATA BC,DE,HL,SP,B,C,D,E,H,L,(HL),A,NZ,Z,NC,C,PO,PE,P,M
845 ' BREAK CHECK ROUTINE
850 S$=INKEY$(0):IF S$<>CHR$(&H30) THEN RETURN
855 T$=INKEY$(1):IF T$="1" THEN T$="":S$="":RETURN ELSE GOTO 855
860 EG=VAL("&H"+MC$)
865 IF EG < &H20 THEN EF$="." ELSE EF$=CHR$(EG)
870 RETURN
875 IF EF=1 THEN Y=BJ ELSE Y=3
880 LOCATE 38,CSRLIN:FOR Z=0 TO Y-1:PRINT E$(Z);:NEXT Z:PRINT E$(Y)
885 RETURN
```

逆アセンブラの特色

① マシン語プログラムは，一般にはプログラム部分とデータ（たとえば，文字列データなど）部分で構成されています（もちろんプログラム部分だけのマシン語プログラムも少なくないのですが）。したがって，逆アセンブルするときは，データ部分をプログラム部分と区別して逆アセンブルしないと，逆アセンブルの結果はメチャクチャになってしまって解読のしょうもないのがほとんどです。

今回作った逆アセンブラは，モード選択によってこの２つの部分を区別して逆アセンブルすることができるようになっています。

例１はHuBASICの&H0F42番地から&H0FE1 番地までの部分をデータとして逆アセンブルした例です。実はこのメモリ範囲はファンクションキーテーブルになっていて，逆アセンブルした結果からもそれがわかります。

例２はHuBASICの&H04BAから&H04C7 までの部分をプログラムとして

逆アセンブルした例です。この部分は" "(スペース)を1個表示させるサブルーチンです。

② プログラム部分の逆アセンブルモード

例 1

```
0F42 0646494C    DB    06H,46H,49H,4CH   .FIL
0F46 45530D00    DB    45H,53H,0DH,00H   ES..
0F4A 00000000    DB    00H,00H,00H,00H   ....
0F4E 00000000    DB    00H,00H,00H,00H   ....
0F52 073F5449    DB    07H,3FH,54H,49H   .?TI
0F56 4D45240D    DB    4DH,45H,24H,0DH   ME$.
0F5A 00000000    DB    00H,00H,00H,00H   ....
0F5E 00000000    DB    00H,00H,00H,00H   ....
0F62 034B4559    DB    03H,4BH,45H,59H   .KEY
0F66 00000000    DB    00H,00H,00H,00H   ....
0F6A 00000000    DB    00H,00H,00H,00H   ....
0F6E 00000000    DB    00H,00H,00H,00H   ....
0F72 064C4953    DB    06H,4CH,49H,53H   .LIS
0F76 541A0D00    DB    54H,1AH,0DH,00H   T...
0F7A 00000000    DB    00H,00H,00H,00H   ....
0F7E 00000000    DB    00H,00H,00H,00H   ....
0F82 0652554E    DB    06H,52H,55H,4EH   .RUN
0F86 20200D00    DB    20H,20H,0DH,00H     ..
0F8A 00000000    DB    00H,00H,00H,00H   ....
0F8E 00000000    DB    00H,00H,00H,00H   ....
0F92 064C4F41    DB    06H,4CH,4FH,41H   .LOA
0F96 44200D00    DB    44H,20H,0DH,00H   D ..
0F9A 00000000    DB    00H,00H,00H,00H   ....
0F9E 00000000    DB    00H,00H,00H,00H   ....
0FA2 06574944    DB    06H,57H,49H,44H   .WID
0FA6 54482000    DB    54H,48H,20H,00H   TH .
0FAA 00000000    DB    00H,00H,00H,00H   ....
0FAE 00000000    DB    00H,00H,00H,00H   ....
0FB2 05434852    DB    05H,43H,48H,52H   .CHR
0FB6 24280000    DB    24H,28H,00H,00H   $(..
0FBA 00000000    DB    00H,00H,00H,00H   ....
0FBE 00000000    DB    00H,00H,00H,00H   ....
0FC2 0650414C    DB    06H,50H,41H,4CH   .PAL
0FC6 45542000    DB    45H,54H,20H,00H   ET .
0FCA 00000000    DB    00H,00H,00H,00H   ....
0FCE 00000000    DB    00H,00H,00H,00H   ....
0FD2 05434F4E    DB    05H,43H,4FH,4EH   .CON
0FD6 540D0000    DB    54H,0DH,00H,00H   T...
0FDA 00000000    DB    00H,00H,00H,00H   ....
0FDE 0000E20F    DB    00H,00H,E2H,0FH   ..♠.
```

では，1列モードと2列モード表示，出力の選択ができます。1列モードにすると逆アセンブルの結果は1列で画面やプリンタに出力されます。また2列モードにすると逆アセンブルの結果は縦2列で出力されます。このモードはプリンタ用紙を2倍に使いたいときに使用してください。

この逆アセンブラでは，出力は62行1ページになっています。したがって2列モードで出力すると，1ページ当たり62×2＝124行分になります。

また，プリンタを使用するときには，FORM LENGTH設定用ロータリスイッチ表示を7番目にセットしてからスイッチオンしてください。このようにすると，1ページにプリントできる行数は66行になり(6LPIの場合)プログラムは62行使うので上と下に2行ずつのスペースが空いて，ちょうどきれいに見えます。プリンタのスイッチオンの前にきちんと用紙の頭の部分を合わせておかないときれいに印字ができません。必ず合わせてからスイッチオンしてください。

例 2

```
04BA 3E20      LD    A,20H
04BC F5        PUSH  AF
04BD C5        PUSH  BC
04BE D5        PUSH  DE
04BF E5        PUSH  HL
04C0 CDD604    CALL  04D6H
04C3 E1        POP   HL
04C4 D1        POP   DE
04C5 C1        POP   BC
04C6 F1        POP   AF
04C7 C9        RET
```

例3はHuBASIC(CZ-8CF01)の&H0FE2から&H12ACまでを1列モードで逆アセンブルした例です。

例4は例3と同じ範囲を2列モードで逆アセンブルした例です。

使い方

図1にプログラムのフローチャートを示します。

例5がプログラムのスタート直後を表わしています。まず，プリンタを使用するかどうかです。使わないときは0[↵]，使うときは1[↵]を押してください。何も押さないで[↵]だけを押すと，プリンタを使わないモードになります。

続いて「スタートアドレス」と「エンドアドレス」を聞いてきます(例7)。ここでは，0〜4ケタの16進数で答えてください。なお，スタートアドレスがエンドアドレスより大きい入力をしますと,何度も聞き返してきます。

次に「モード」を聞いてきます。マシン語プログラムとみなして逆アセンブルして良い場合は「1」を，データとしての場合は「2」を入力してください。何も入力しないで[↵]すると「1」モードで実行します。「1」を選ぶと，何列まで印字するかを聞いてきます。1列のときは「1」を，2列のときは「2」を入力して[↵]してください。何も入力しないで[↵]すると，1例モードで実行します。また，モード選択で「2」を選ぶとただちに実行されます。

例6は，プリンタは使わず「0」&H0000番地から&H0100番地まで，逆アセンブル・モード「1」で2列「指定 したとこところです。

例 3

```
0FE2 E5          PUSH HL
0FE3 ED5B7E14    LD   DE,(147EH)
0FE7 D5          PUSH DE
0FE8 112D10      LD   DE,102DH
0FEB ED537E14    LD   (147EH),DE
0FEF 3A0600      LD   A,(0006H)
0FF2 F5          PUSH AF
0FF3 3E50        LD   A,50H
0FF5 320600      LD   (0006H),A
0FF8 ED73F110    LD   (10F1H),SP
0FFC AF          XOR  A
```

例 4

```
0FE2 E5          PUSH HL               1056 6A          LD   L,D
0FE3 ED5B7E14    LD   DE,(147EH)       1057 104C        DJNZ 10A5H
0FE7 D5          PUSH DE               1059 9A          SBC  A,D
0FE8 112D10      LD   DE,102DH         105A 1056        DJNZ 10B2H
0FEB ED537E14    LD   (147EH),DE       105C E1          POP  HL
0FEF 3A0600      LD   A,(0006H)        105D 1054        DJNZ 10B3H
0FF2 F5          PUSH AF               105F BE          CP   (HL)
0FF3 3E50        LD   A,50H            1060 12          LD   (DE),A
0FF5 320600      LD   (0006H),A        1061 3A7214      LD   A,(1472H)
0FF8 ED73F110    LD   (10F1H),SP       1064 EE01        XOR  01H
0FFC AF          XOR  A                1066 327214      LD   (1472H),A
```

プログラムの実行が終わると，さらに続けるかどうかを聞いてきます。続けるときは1 ↵ を入力してください。この場合，プリンタ行カウンタ用変数は保存されるため，以後もその紙の62行目で改行されます。また，プリンタを使わないモードを指定する場合は，プリンタ行カウンタ用変数も変わりません。実際にプリンタに出力する前に画面で見てから，というようなことができるので便利です。

なお，続けないときには「1」以外の数を入力してください。このときもう1回RUNすると，プリンタ行カウンタ変数が「1」に再設定されるので，注意してください。また，実行(逆アセンブル)している途中で一時的に止めたいときには「0」を．続けるときは「1」を押してください。

もうひとつ，説明するのを忘れてしまいましたが，逆アセンブルしたいプログラムがメインメモリの上位アドレス（このプログラムより上位アドレス）にある場合は，マシン語プログラムを保護するため，プログラムをRUNする前に「CLEAR」文を行なってからRUNするようにしてください。「CLEAR」文の使い方については，マニュアルを見てください。

また，このプログラムではカラー機能をかなり使っており，画面で実行の結果が見やすいように工夫してあります。具体的には，アセンブル・モードの実行が緑色で，データ・モードの実行は青色，パラメータ入力は赤色となっています。色だけ見ていてもけっこう楽しいものです。このプログラムはBASICのごく普通の命令しか使っていないので，他の機種への移植もかんたんにできると思います。最後に，この逆アセンブラを使って，Z-80の命令リストを作りましたので参考にしてください。

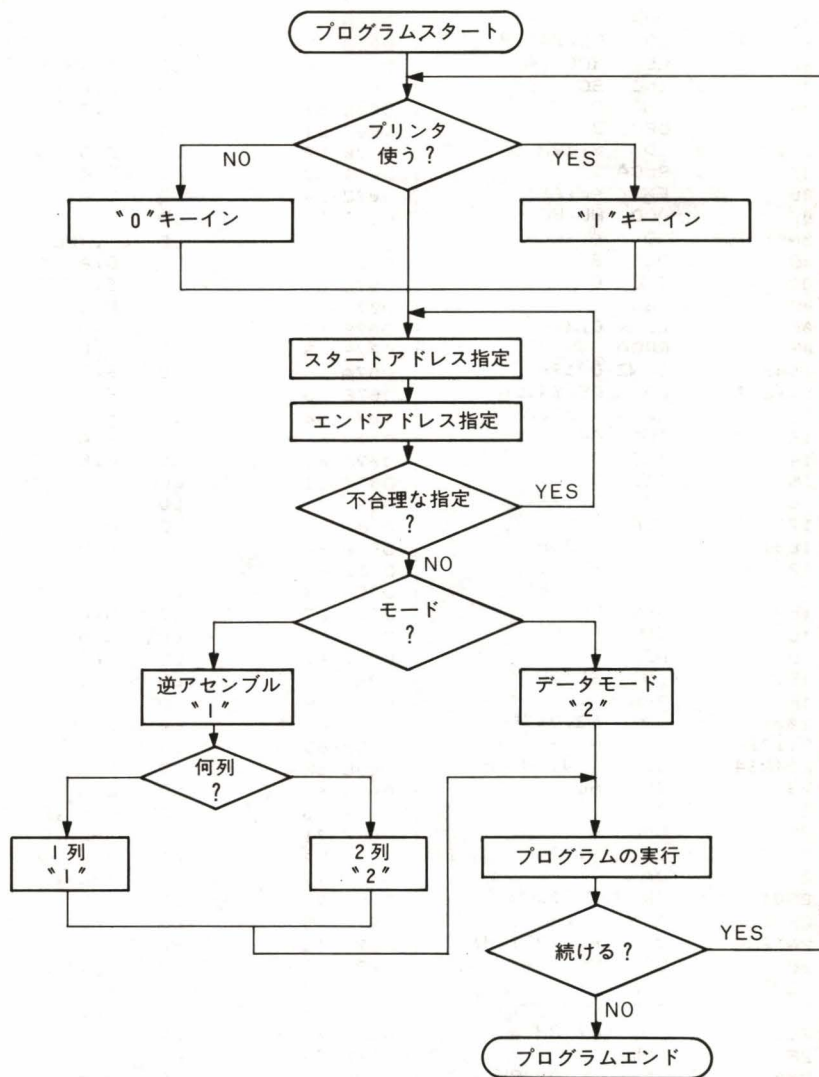



図1

プログラムスタート
プリンタ使う？
NO
YES
"0"キーイン
"1"キーイン
スタートアドレス指定
エンドアドレス指定
不合理な指定？
YES
NO
モード？
逆アセンブル "1"
データモード "2"
何列？
1列 "1"
2列 "2"
プログラムの実行
続ける？
YES
NO
プログラムエンド

例5

```
Z-80 DISASSEMBLER for X1
by Itti Rittaporn
1983 APRIL 29

PRINTER  ONLINE...  1
         OFFLINE.. [0]
```

例6

```
Z-80 DISASSEMBLER for X1
by Itti Rittaporn
1983 APRIL 29

PRINTER  ONLINE...  1
         OFFLINE.. [0] 0

start address =  0000
end   address =  0100

Disassembler mode..[1]
Data         mode...2  1

ONE ROW..[1]
TWO ROWS..2  2
```

Z-80命令リスト

前号の「Z-80マシン語→アセンブリ対照表」,「マシン語早見表」とあわせてご活用ください)

```
D000 00        NOP
D001 011234    LD    BC,3412H
D004 02        LD    (BC),A
D005 03        INC   BC
D006 04        INC   B
D007 05        DEC   B
D008 0612      LD    B,12H
D00A 07        RLCA
D00B 08        EX    AF,AF'
D00C 09        ADD   HL,BC
D00D 0A        LD    A,(BC)
D00E 0B        DEC   BC
D00F 0C        INC   C
D010 0D        DEC   C
D011 0E12      LD    C,12H
D013 0F        RRCA
D014 1003      DJNZ  D019H
D016 111234    LD    DE,3412H
D019 12        LD    (DE),A
D01A 13        INC   DE
D01B 14        INC   D
D01C 15        DEC   D
D01D 1612      LD    D,12H
D01F 17        RLA
D020 1801      JR    D023H
D022 19        ADD   HL,DE
D023 1A        LD    A,(DE)
D024 1B        DEC   DE
D025 1C        INC   E
D026 1D        DEC   E
D027 1E12      LD    E,12H
D029 1F        RRA
D02A 2003      JR    NZ,D02FH
D02C 211234    LD    HL,3412H
D02F 221234    LD    (3412H),HL
D032 23        INC   HL
D033 24        INC   H
D034 25        DEC   H
D035 2612      LD    H,12H
D037 27        DAA
D038 2801      JR    Z,D03BH
D03A 29        ADD   HL,HL
D03B 2A1234    LD    HL,(3412H)
D03E 2B        DEC   HL
D03F 2C        INC   L
D040 2D        DEC   L
D041 2E12      LD    L,12H
D043 2F        CPL
D044 3003      JR    NC,D049H
D046 311234    LD    SP,3412H
D049 321234    LD    (3412H),A
D04C 33        INC   SP
D04D 34        INC   (HL)
D04E 35        DEC   (HL)
D04F 3612      LD    (HL),12H
D051 37        SCF
D052 3801      JR    C,D055H
D054 39        ADD   HL,SP
D055 3A1234    LD    A,(3412H)
D058 3B        DEC   SP
D059 3C        INC   A
D05A 3D        DEC   A
D05B 3E12      LD    A,12H
D05D 3F        CCF
D05E 40        LD    B,B
D05F 41        LD    B,C
D060 42        LD    B,D
D061 43        LD    B,E
D062 44        LD    B,H
D063 45        LD    B,L
D064 46        LD    B,(HL)
D065 47        LD    B,A
D066 48        LD    C,B
D067 49        LD    C,C
D068 4A        LD    C,D
D069 4B        LD    C,E
D06A 4C        LD    C,H
D06B 4D        LD    C,L
D06C 4E        LD    C,(HL)
D06D 4F        LD    C,A
D06E 50        LD    D,B
D06F 51        LD    D,C
D070 52        LD    D,D
D071 53        LD    D,E
D072 54        LD    D,H
D073 55        LD    D,L
D074 56        LD    D,(HL)
D075 57        LD    D,A
D076 58        LD    E,B
D077 59        LD    E,C
D078 5A        LD    E,D
D079 5B        LD    E,E
D07A 5C        LD    E,H
D07B 5D        LD    E,L
D07C 5E        LD    E,(HL)
D07D 5F        LD    E,A
D07E 60        LD    H,B
D07F 61        LD    H,C
D080 62        LD    H,D
D081 63        LD    H,E
D082 64        LD    H,H
D083 65        LD    H,L
D084 66        LD    H,(HL)
D085 67        LD    H,A
D086 68        LD    L,B
D087 69        LD    L,C
D088 6A        LD    L,D
D089 6B        LD    L,E
D08A 6C        LD    L,H
D08B 6D        LD    L,L
D08C 6E        LD    L,(HL)
D08D 6F        LD    L,A
D08E 70        LD    (HL),B
D08F 71        LD    (HL),C
D090 72        LD    (HL),D
D091 73        LD    (HL),E
D092 74        LD    (HL),H
D093 75        LD    (HL),L
D094 76        HALT
D095 77        LD    (HL),A
D096 78        LD    A,B
D097 79        LD    A,C
D098 7A        LD    A,D
D099 7B        LD    A,E
D09A 7C        LD    A,H
D09B 7D        LD    A,L
D09C 7E        LD    A,(HL)
D09D 7F        LD    A,A
D09E 80        ADD   A,B
D09F 81        ADD   A,C
D0A0 82        ADD   A,D
D0A1 83        ADD   A,E
D0A2 84        ADD   A,H
D0A3 85        ADD   A,L
D0A4 86        ADD   A,(HL)
D0A5 87        ADD   A,A
D0A6 88        ADC   A,B
D0A7 89        ADC   A,C
D0A8 8A        ADC   A,D
D0A9 8B        ADC   A,E
D0AA 8C        ADC   A,H
D0AB 8D        ADC   A,L
D0AC 8E        ADC   A,(HL)
D0AD 8F        ADC   A,A
D0AE 90        SUB   B
D0AF 91        SUB   C
D0B0 92        SUB   D
D0B1 93        SUB   E
D0B2 94        SUB   H
D0B3 95        SUB   L
D0B4 96        SUB   (HL)
D0B5 97        SUB   A
D0B6 98        SBC   A,B
D0B7 99        SBC   A,C
D0B8 9A        SBC   A,D
D0B9 9B        SBC   A,E
D0BA 9C        SBC   A,H
D0BB 9D        SBC   A,L
D0BC 9E        SBC   A,(HL)
D0BD 9F        SBC   A,A
D0BE A0        AND   B
D0BF A1        AND   C
D0C0 A2        AND   D
D0C1 A3        AND   E
D0C2 A4        AND   H
D0C3 A5        AND   L
D0C4 A6        AND   (HL)
D0C5 A7        AND   A
D0C6 A8        XOR   B
D0C7 A9        XOR   C
D0C8 AA        XOR   D
D0C9 AB        XOR   E
D0CA AC        XOR   H
D0CB AD        XOR   L
D0CC AE        XOR   (HL)
D0CD AF        XOR   A
D0CE B0        OR    B
D0CF B1        OR    C
D0D0 B2        OR    D
D0D1 B3        OR    E
D0D2 B4        OR    H
D0D3 B5        OR    L
D0D4 B6        OR    (HL)
D0D5 B7        OR    A
D0D6 B8        CP    B
D0D7 B9        CP    C
D0D8 BA        CP    D
D0D9 BB        CP    E
D0DA BC        CP    H
D0DB BD        CP    L
D0DC BE        CP    (HL)
D0DD BF        CP    A
D0DE C0        RET   NZ
D0DF C1        POP   BC
D0E0 C21234    JP    NZ,3412H
D0E3 C31234    JP    3412H
D0E6 C41234    CALL  NZ,3412H
D0E9 C5        PUSH  BC
D0EA C612      ADD   A,12H
D0EC C7        RST   00H
D0ED C8        RET   Z
D0EE C9        RET
D0EF CA1234    JP    Z,3412H
D0F2 CB00      RLC   B
D0F4 CB01      RLC   C
D0F6 CB02      RLC   D
D0F8 CB03      RLC   E
D0FA CB04      RLC   H
D0FC CB05      RLC   L
D0FE CB06      RLC   (HL)
D100 CB07      RLC   A
D102 CB08      RRC   B
D104 CB09      RRC   C
D106 CB0A      RRC   D
D108 CB0B      RRC   E
D10A CB0C      RRC   H
D10C CB0D      RRC   L
D10E CB0E      RRC   (HL)
D110 CB0F      RRC   A
D112 CB10      RL    B
D114 CB11      RL    C
D116 CB12      RL    D
D118 CB13      RL    E
D11A CB14      RL    H
D11C CB15      RL    L
D11E CB16      RL    (HL)
D120 CB17      RL    A
D122 CB18      RR    B
```

```
D124 CB19       RR   C
D126 CB1A       RR   D
D128 CB1B       RR   E
D12A CB1C       RR   H
D12C CB1D       RR   L
D12E CB1E       RR   (HL)
D130 CB1F       RR   A
D132 CB20       SLA  B
D134 CB21       SLA  C
D136 CB22       SLA  D
D138 CB23       SLA  E
D13A CB24       SLA  H
D13C CB25       SLA  L
D13E CB26       SLA  (HL)
D140 CB27       SLA  A
D142 CB28       SRA  B
D144 CB29       SRA  C
D146 CB2A       SRA  D
D148 CB2B       SRA  E
D14A CB2C       SRA  H
D14C CB2D       SRA  L
D14E CB2E       SRA  (HL)
D150 CB2F       SRA  A
D152 CB38       SRL  B
D154 CB39       SRL  C
D156 CB3A       SRL  D
D158 CB3B       SRL  E
D15A CB3C       SRL  H
D15C CB3D       SRL  L
D15E CB3E       SRL  (HL)
D160 CB3F       SRL  A
D162 CB40       BIT  0,B
D164 CB41       BIT  0,C
D166 CB42       BIT  0,D
D168 CB43       BIT  0,E
D16A CB44       BIT  0,H
D16C CB45       BIT  0,L
D16E CB46       BIT  0,(HL)
D170 CB47       BIT  0,A
D172 CB48       BIT  1,B
D174 CB49       BIT  1,C
D176 CB4A       BIT  1,D
D178 CB4B       BIT  1,E
D17A CB4C       BIT  1,H
D17C CB4D       BIT  1,L
D17E CB4E       BIT  1,(HL)
D180 CB4F       BIT  1,A
D182 CB50       BIT  2,B
D184 CB51       BIT  2,C
D186 CB52       BIT  2,D
D188 CB53       BIT  2,E
D18A CB54       BIT  2,H
D18C CB55       BIT  2,L
D18E CB56       BIT  2,(HL)
D190 CB57       BIT  2,A
D192 CB58       BIT  3,B
D194 CB59       BIT  3,C
D196 CB5A       BIT  3,D
D198 CB5B       BIT  3,E
D19A CB5C       BIT  3,H
D19C CB5D       BIT  3,L
D19E CB5E       BIT  3,(HL)
D1A0 CB5F       BIT  3,A
D1A2 CB60       BIT  4,B
D1A4 CB61       BIT  4,C
D1A6 CB62       BIT  4,D
D1A8 CB63       BIT  4,E
D1AA CB64       BIT  4,H
D1AC CB65       BIT  4,L
D1AE CB66       BIT  4,(HL)
D1B0 CB67       BIT  4,A
D1B2 CB68       BIT  5,B
D1B4 CB69       BIT  5,C
D1B6 CB6A       BIT  5,D
D1B8 CB6B       BIT  5,E
D1BA CB6C       BIT  5,H
D1BC CB6D       BIT  5,L
D1BE CB6E       BIT  5,(HL)
```

```
D1C0 CB6F       BIT  5,A
D1C2 CB70       BIT  6,B
D1C4 CB71       BIT  6,C
D1C6 CB72       BIT  6,D
D1C8 CB73       BIT  6,E
D1CA CB74       BIT  6,H
D1CC CB75       BIT  6,L
D1CE CB76       BIT  6,(HL)
D1D0 CB77       BIT  6,A
D1D2 CB78       BIT  7,B
D1D4 CB79       BIT  7,C
D1D6 CB7A       BIT  7,D
D1D8 CB7B       BIT  7,E
D1DA CB7C       BIT  7,H
D1DC CB7D       BIT  7,L
D1DE CB7E       BIT  7,(HL)
D1E0 CB7F       BIT  7,A
D1E2 CB80       RES  0,B
D1E4 CB81       RES  0,C
D1E6 CB82       RES  0,D
D1E8 CB83       RES  0,E
D1EA CB84       RES  0,H
D1EC CB85       RES  0,L
D1EE CB86       RES  0,(HL)
D1F0 CB87       RES  0,A
D1F2 CB88       RES  1,B
D1F4 CB89       RES  1,C
D1F6 CB8A       RES  1,D
D1F8 CB8B       RES  1,E
D1FA CB8C       RES  1,H
D1FC CB8D       RES  1,L
D1FE CB8E       RES  1,(HL)
D200 CB8F       RES  1,A
D202 CB90       RES  2,B
D204 CB91       RES  2,C
D206 CB92       RES  2,D
D208 CB93       RES  2,E
D20A CB94       RES  2,H
D20C CB95       RES  2,L
D20E CB96       RES  2,(HL)
D210 CB97       RES  2,A
D212 CB98       RES  3,B
D214 CB99       RES  3,C
D216 CB9A       RES  3,D
D218 CB9B       RES  3,E
D21A CB9C       RES  3,H
D21C CB9D       RES  3,L
D21E CB9E       RES  3,(HL)
D220 CB9F       RES  3,A
D222 CBA0       RES  4,B
D224 CBA1       RES  4,C
D226 CBA2       RES  4,D
D228 CBA3       RES  4,E
D22A CBA4       RES  4,H
D22C CBA5       RES  4,L
D22E CBA6       RES  4,(HL)
D230 CBA7       RES  4,A
D232 CBA8       RES  5,B
D234 CBA9       RES  5,C
D236 CBAA       RES  5,D
D238 CBAB       RES  5,E
D23A CBAC       RES  5,H
D23C CBAD       RES  5,L
D23E CBAE       RES  5,(HL)
D240 CBAF       RES  5,A
D242 CBB0       RES  6,B
D244 CBB1       RES  6,C
D246 CBB2       RES  6,D
D248 CBB3       RES  6,E
D24A CBB4       RES  6,H
D24C CBB5       RES  6,L
D24E CBB6       RES  6,(HL)
D250 CBB7       RES  6,A
D252 CBB8       RES  7,B
D254 CBB9       RES  7,C
D256 CBBA       RES  7,D
D258 CBBB       RES  7,E
D25A CBBC       RES  7,H
```

```
D25C CBBD       RES  7,L
D25E CBBE       RES  7,(HL)
D260 CBBF       RES  7,A
D262 CBC0       SET  0,B
D264 CBC1       SET  0,C
D266 CBC2       SET  0,D
D268 CBC3       SET  0,E
D26A CBC4       SET  0,H
D26C CBC5       SET  0,L
D26E CBC6       SET  0,(HL)
D270 CBC7       SET  0,A
D272 CBC8       SET  1,B
D274 CBC9       SET  1,C
D276 CBCA       SET  1,D
D278 CBCB       SET  1,E
D27A CBCC       SET  1,H
D27C CBCD       SET  1,L
D27E CBCE       SET  1,(HL)
D280 CBCF       SET  1,A
D282 CBD0       SET  2,B
D284 CBD1       SET  2,C
D286 CBD2       SET  2,D
D288 CBD3       SET  2,E
D28A CBD4       SET  2,H
D28C CBD5       SET  2,L
D28E CBD6       SET  2,(HL)
D290 CBD7       SET  2,A
D292 CBD8       SET  3,B
D294 CBD9       SET  3,C
D296 CBDA       SET  3,D
D298 CBDB       SET  3,E
D29A CBDC       SET  3,H
D29C CBDD       SET  3,L
D29E CBDE       SET  3,(HL)
D2A0 CBDF       SET  3,A
D2A2 CBE0       SET  4,B
D2A4 CBE1       SET  4,C
D2A6 CBE2       SET  4,D
D2A8 CBE3       SET  4,E
D2AA CBE4       SET  4,H
D2AC CBE5       SET  4,L
D2AE CBE6       SET  4,(HL)
D2B0 CBE7       SET  4,A
D2B2 CBE8       SET  5,B
D2B4 CBE9       SET  5,C
D2B6 CBEA       SET  5,D
D2B8 CBEB       SET  5,E
D2BA CBEC       SET  5,H
D2BC CBED       SET  5,L
D2BE CBEE       SET  5,(HL)
D2C0 CBEF       SET  5,A
D2C2 CBF0       SET  6,B
D2C4 CBF1       SET  6,C
D2C6 CBF2       SET  6,D
D2C8 CBF3       SET  6,E
D2CA CBF4       SET  6,H
D2CC CBF5       SET  6,L
D2CE CBF6       SET  6,(HL)
D2D0 CBF7       SET  6,A
D2D2 CBF8       SET  7,B
D2D4 CBF9       SET  7,C
D2D6 CBFA       SET  7,D
D2D8 CBFB       SET  7,E
D2DA CBFC       SET  7,H
D2DC CBFD       SET  7,L
D2DE CBFE       SET  7,(HL)
D2E0 CBFF       SET  7,A
D2E2 CC1234     CALL Z,3412H
D2E5 CD1234     CALL 3412H
D2E8 CE12       ADC  A,12H
D2EA CF         RST  08H
D2EB D0         RET  NC
D2EC D1         POP  DE
D2ED D21234     JP   NC,3412H
D2F0 D312       OUT  (12H),A
D2F2 D41234     CALL NC,3412H
D2F5 D5         PUSH DE
D2F6 D612       SUB  12H
```

```
D2F8 D7         RST   10H
D2F9 D8         RET   C
D2FA D9         EXX
D2FB DA1234     JP    C,3412H
D2FE DB12       IN    A,(12H)
D300 DC1234     CALL  C,3412H
D303 DD09       ADD   IX,BC
D305 DD19       ADD   IX,DE
D307 DD211234   LD    IX,3412H
D30B DD221234   LD    (3412H),IX
D30F DD23       INC   IX
D311 DD29       ADD   IX,IX
D313 DD2A1234   LD    IX,(3412H)
D317 DD2B       DEC   IX
D319 DD3412     INC   (IX+12H)
D31C DD3512     DEC   (IX+12H)
D31F DD361234   LD    (IX+12H),34H
D323 DD39       ADD   IX,SP
D325 DD4612     LD    B,(IX+12H)
D328 DD4E12     LD    C,(IX+12H)
D32B DD5612     LD    D,(IX+12H)
D32E DD5E12     LD    E,(IX+12H)
D331 DD6612     LD    H,(IX+12H)
D334 DD6E12     LD    L,(IX+12H)
D337 DD7012     LD    (IX+12H),B
D33A DD7112     LD    (IX+12H),C
D33D DD7212     LD    (IX+12H),D
D340 DD7312     LD    (IX+12H),E
D343 DD7412     LD    (IX+12H),H
D346 DD7512     LD    (IX+12H),L
D349 DD7712     LD    (IX+12H),A
D34C DD7E12     LD    A,(IX+12H)
D34F DD8612     ADD   A,(IX+12H)
D352 DD8E12     ADC   A,(IX+12H)
D355 DD9612     SUB   (IX+12H)
D358 DD9E12     SBC   A,(IX+12H)
D35B DDA612     AND   (IX+12H)
D35E DDAE12     XOR   (IX+12H)
D361 DDB612     OR    (IX+12H)
D364 DDBE12     CP    (IX+12H)
D367 DDE1       POP   IX
D369 DDE3       EX    (SP),IX
D36B DDE5       PUSH  IX
D36D DDE9       JP    (IX)
D36F DDF9       LD    SP,IX
D371 DDCB1206   RLC   (IX+12H)
D375 DDCB120E   RRC   (IX+12H)
D379 DDCB1216   RL    (IX+12H)
D37D DDCB121E   RR    (IX+12H)
D381 DDCB1226   SLA   (IX+12H)
D385 DDCB122E   SRA   (IX+12H)
D389 DDCB123E   SRL   (IX+12H)
D38D DDCB1246   BIT   0,(IX+12H)
D391 DDCB124E   BIT   1,(IX+12H)
D395 DDCB1256   BIT   2,(IX+12H)
D399 DDCB125E   BIT   3,(IX+12H)
D39D DDCB1266   BIT   4,(IX+12H)
D3A1 DDCB126E   BIT   5,(IX+12H)
D3A5 DDCB1276   BIT   6,(IX+12H)
D3A9 DDCB127E   BIT   7,(IX+12H)
D3AD DDCB1286   RES   0,(IX+12H)
D3B1 DDCB128E   RES   1,(IX+12H)
D3B5 DDCB1296   RES   2,(IX+12H)
D3B9 DDCB129E   RES   3,(IX+12H)
D3BD DDCB12A6   RES   4,(IX+12H)
D3C1 DDCB12AE   RES   5,(IX+12H)
D3C5 DDCB12B6   RES   6,(IX+12H)
D3C9 DDCB12BE   RES   7,(IX+12H)
D3CD DDCB12C6   SET   0,(IX+12H)
D3D1 DDCB12CE   SET   1,(IX+12H)
D3D5 DDCB12D6   SET   2,(IX+12H)
D3D9 DDCB12DE   SET   3,(IX+12H)
D3DD DDCB12E6   SET   4,(IX+12H)
D3E1 DDCB12EE   SET   5,(IX+12H)
D3E5 DDCB12F6   SET   6,(IX+12H)
D3E9 DDCB12FE   SET   7,(IX+12H)
D3ED DE12       SBC   A,12H
D3EF DF         RST   18H
D3F0 E0         RET   PO
D3F1 E1         POP   HL
D3F2 E21234     JP    PO,3412H
D3F5 E3         EX    (SP),HL
D3F6 E41234     CALL  PO,3412H
D3F9 E5         PUSH  HL
D3FA E612       AND   12H
D3FC E7         RST   20H
D3FD E8         RET   PE
D3FE E9         JP    (HL)
D3FF EA1234     JP    PE,3412H
D402 EB         EX    DE,HL
D403 EC1234     CALL  PE,3412H
D406 ED40       IN    B,(C)
D408 ED41       OUT   (C),B
D40A ED42       SBC   HL,BC
D40C ED431234   LD    (3412H),BC
D410 ED44       NEG
D412 ED45       RETN
D414 ED46       IM    0
D416 ED47       LD    I,A
D418 ED48       IN    C,(C)
D41A ED49       OUT   (C),C
D41C ED4A       ADC   HL,BC
D41E ED4B1234   LD    BC,(3412H)
D422 ED4D       RETI
D424 ED50       IN    D,(C)
D426 ED51       OUT   (C),D
D428 ED52       SBC   HL,DE
D42A ED531234   LD    (3412H),DE
D42E ED56       IM    1
D430 ED57       LD    A,I
D432 ED58       IN    E,(C)
D434 ED59       OUT   (C),E
D436 ED5A       ADC   HL,DE
D438 ED5B1234   LD    DE,(3412H)
D43C ED5E       IM    2
D43E ED60       IN    H,(C)
D440 ED61       OUT   (C),H
D442 ED62       SBC   HL,HL
D444 ED67       RRD
D446 ED68       IN    L,(C)
D448 ED69       OUT   (C),L
D44A ED6A       ADC   HL,HL
D44C ED6F       RLD
D44E ED72       SBC   HL,SP
D450 ED731234   LD    (3412H),SP
D454 ED78       IN    A,(C)
D456 ED79       OUT   (C),A
D458 ED7A       ADC   HL,SP
D45A ED7B1234   LD    SP,(3412H)
D45E EDA0       LDI
D460 EDA1       CPI
D462 EDA2       INI
D464 EDA3       OUTI
D466 EDA8       LDD
D468 EDA9       CPD
D46A EDAA       IND
D46C EDAB       OUTD
D46E EDB0       LDIR
D470 EDB1       CPIR
D472 EDB2       INIR
D474 EDB3       OTIR
D476 EDB8       LDDR
D478 EDB9       CPDR
D47A EDBA       INDR
D47C EDBB       OTDR
D47E EE12       XOR   12H
D480 EF         RST   28H
D481 F0         RET   P
D482 F1         POP   AF
D483 F21234     JP    P,3412H
D486 F3         DI
D487 F41234     CALL  P,3412H
D48A F5         PUSH  AF
D48B F612       OR    12H
D48D F7         RST   30H
D48E F8         RET   M
D48F F9         LD    SP,HL
D490 FA1234     JP    M,3412H
D493 FB         EI
D494 FC1234     CALL  M,3412H
D497 FD09       ADD   IY,BC
D499 FD19       ADD   IY,DE
D49B FD211234   LD    IY,3412H
D49F FD221234   LD    (3412H),IY
D4A3 FD23       INC   IY
D4A5 FD29       ADD   IY,IY
D4A7 FD2A1234   LD    IY,(3412H)
D4AB FD2B       DEC   IY
D4AD FD3412     INC   (IY+12H)
D4B0 FD3512     DEC   (IY+12H)
D4B3 FD361234   LD    (IY+12H),34H
D4B7 FD39       ADD   IY,SP
D4B9 FD4612     LD    B,(IY+12H)
D4BC FD4E12     LD    C,(IY+12H)
D4BF FD5612     LD    D,(IY+12H)
D4C2 FD5E12     LD    E,(IY+12H)
D4C5 FD6612     LD    H,(IY+12H)
D4C8 FD6E12     LD    L,(IY+12H)
D4CB FD7012     LD    (IY+12H),B
D4CE FD7112     LD    (IY+12H),C
D4D1 FD7212     LD    (IY+12H),D
D4D4 FD7312     LD    (IY+12H),E
D4D7 FD7412     LD    (IY+12H),H
D4DA FD7512     LD    (IY+12H),L
D4DD FD7712     LD    (IY+12H),A
D4E0 FD7E12     LD    A,(IY+12H)
D4E3 FD8612     ADD   A,(IY+12H)
D4E6 FD8E12     ADC   A,(IY+12H)
D4E9 FD9612     SUB   (IY+12H)
D4EC FD9E12     SBC   A,(IY+12H)
D4EF FDA612     AND   (IY+12H)
D4F2 FDAE12     XOR   (IY+12H)
D4F5 FDB612     OR    (IY+12H)
D4F8 FDBE12     CP    (IY+12H)
D4FB FDE1       POP   IY
D4FD FDE3       EX    (SP),IY
D4FF FDE5       PUSH  IY
D501 FDE9       JP    (IY)
D503 FDF9       LD    SP,IY
D505 FDCB1206   RLC   (IY+12H)
D509 FDCB120E   RRC   (IY+12H)
D50D FDCB1216   RL    (IY+12H)
D511 FDCB121E   RR    (IY+12H)
D515 FDCB1226   SLA   (IY+12H)
D519 FDCB122E   SRA   (IY+12H)
D51D FDCB123E   SRL   (IY+12H)
D521 FDCB1246   BIT   0,(IY+12H)
D525 FDCB124E   BIT   1,(IY+12H)
D529 FDCB1256   BIT   2,(IY+12H)
D52D FDCB125E   BIT   3,(IY+12H)
D531 FDCB1266   BIT   4,(IY+12H)
D535 FDCB126E   BIT   5,(IY+12H)
D539 FDCB1276   BIT   6,(IY+12H)
D53D FDCB127E   BIT   7,(IY+12H)
D541 FDCB1286   RES   0,(IY+12H)
D545 FDCB128E   RES   1,(IY+12H)
D549 FDCB1296   RES   2,(IY+12H)
D54D FDCB129E   RES   3,(IY+12H)
D551 FDCB12A6   RES   4,(IY+12H)
D555 FDCB12AE   RES   5,(IY+12H)
D559 FDCB12B6   RES   6,(IY+12H)
D55D FDCB12BE   RES   7,(IY+12H)
D561 FDCB12C6   SET   0,(IY+12H)
D565 FDCB12CE   SET   1,(IY+12H)
D569 FDCB12D6   SET   2,(IY+12H)
D56D FDCB12DE   SET   3,(IY+12H)
D571 FDCB12E6   SET   4,(IY+12H)
D575 FDCB12EE   SET   5,(IY+12H)
D579 FDCB12F6   SET   6,(IY+12H)
D57D FDCB12FE   SET   7,(IY+12H)
D581 FE12       CP    12H
D583 FF         RST   38H
```

東大No.1の省師

コンピュータ マージャン必勝作戦

福田浩介

連載第7回 MZ-80K/C/1200 700(Hu-BASIC)

1 麻雀の得点計算

最近はマイコン麻雀ゲームというのが出回っていて，1人ゲームから4人ゲームまで，さまざまなものがあります。しかも初期のものではアガリ点を何符何翻と入力しなければばならなかったのが，最近ではアガルと自動的に役を数えて得点計算をしてくれるのです。しかし，実際には案外その点数計算も細かいところまでいきとどかず，ちゃんと数えると点数が違っていた，ということもざらにあるようです。

一方,「麻雀を覚えたいが，点数計算が難しそうだから……」とか「麻雀仲間に点数を知らないといってバカにされた」とかいう人も大勢いるでしょう。そういう人は麻雀ゲームの点数計算が違っていても気付かずに済んでしまうのです。実際，麻雀の中級者といわれる人でも点数計算をまちがえることもままありますし，点数計算の問題集もあるくらいですから，かなりいろいろな例外が潜んでいて，それらすべてを知るにはある程度の期間が必要です。

そこで今回，私が作った「完璧ともいえる麻雀点数計算プログラム」を発表いたします。本当は「完璧なる……」にしたかったのですが,「ともいえる」としたのは決して自信がなかったからではなく，次の理由によるのです。

麻雀は古くからあるゲームですが，いまだ社会的地位は低いものです。囲碁・将棋などと比べても，統一的なルールも完成しておらず，地域ごとにさまざまなルールで行なわれているのが現状です。ですから，役についても採用されているのがまちまちなわけです。

そこで今回は，一般的な競技マージャンのルールに近いものにしてみました。競技マージャンといっても，基本的にはごく普通の巷のルールと同じです。

今回のプログラムにそれ以外の役を付け加えることも実際にはかんたんですので，それができるようにプログラムの説明をしていきますが，その前に……。そう，点数計算を知らない人のために，その概要をまとめてみましょう。

2 実際の計算

麻雀の得点計算は，手続きとしては非常にかんたんです。まずアガリの形に対して基礎点というのが与えられます(これを符，もしくは基礎符と言います)。次にその点に役の数だけ2をかけます。この2つが本来の得点計算の姿なのです。しかし，現在ではルールも複雑になり，やっかいな枝葉がついてしまいました。それをひとつひとつ説明するのはその方面の専門書に任せるとして，ここでは図1のアガリで説明してみましょう。

まず，基本となるアガリの形に与えられる点を出します。アガった時点で，基本点20点が与えられます。これに，八万の暗刻に刻子として4点が与えられ，アガリのところで，単騎待ちに2点，さらにツモったことで2点が加わり，計28点となります。ここで1の位は切り上げとなり，結局基礎点は30点（30符）となりました。詳しい点のつけ方はここでは説明しませんが，基本点は元の20点に次の3つの観点から加算されることだけ頭においておいてください。

①刻子・対子の種類によって与えられるもの（槓子も含む)。

②待ちの形によって与えられるもの。

③ツモや面前のロンに与えられるもの。

さて，今度は役の数を数えます（役も本で覚えてください)。面前ツモが1つ，タンヤオが1つ，イーペーコーが1つ，計3つです。このほかに，現在ではあらかじめ最初から2つ余計な役が与えられていて，計5つということになります。これで，得点は30×2×2×2×2×2＝960点，10の位を切り上げて1,000点となります。このアガリが子のものなら子から1,000点ずつ，親からは倍の2,000点で，計4,000点を得ることになります。これが親のアガリだと，すべて倍の2,000点ずつが子からもらえ，計6,000点の収入になります。

さて，先ほどの余計な2つ分は2ゾロといって，常に2役分すなわち4倍されるわけです。しかし役にも1役で2～6つ分の役がつくものもあり，点が膨大な数になるため，8,000点を超えると満貫といって，役の数（翻数(ファン)）で点の打ち切りがあります。あらかじめの2役（二翻）を含めて七翻までで8,000点以上なら満貫で8,000点，八，九翻ならハネ満で12,000点，という具合になるわけです。一応，計算の概要は以上の

ようになっています。

実戦にたけてくると，以上の途中は省略して，30符３翻は1,000，2,000，70符１翻は2,300などと覚えてしまえるわけですが，まあそれまでは計算からマスターするようにしましょう。その間の助けとして，計算ミスのないこのプログラムができたわけです。以上でやってきた手続きをすべてやってくれるプログラムにしようと欲ばって作りました。

③ プログラムについて

点数を計算するために必要な手続きは次のようになります。

まず，アガリの牌14枚を４面子と１雀頭に直します（七対子と国士無双は別）。次に役満であるかどうかを見ます。役満とは，その役ひとつで満貫の４倍，すなわち，子なら32,000点，親で48,000点になるという役です。ここで役満であれば点数がわかりますから得点の表示をします。役満でなければ，今度は符を数え，役を数えて点数を出すわけです。

さて，今回のプログラムで苦労したポイントがあります。それは，同じ和りでも面子のとり方によって，また和り牌をどれにするかによって，得点が違ってくるということです。こういう場合，麻雀では最高値でとることになっています。

まず面子のとり方によって得点が違う例ですが，図２のようなものがあります。これは五索を雀頭にとると，タンヤオ・ピンフ・イーペーコー・ツモで，子の1,300，2,600ですが，二索を雀頭にとれば，タンヤオ・サンショク・ツモで，子の2,000，3,900になります。また，和り牌の違いによる差は図３のようなものです。これは，両面の三万とみると40符１翻の1,300点ですが，カンチャンの三万とみると50符１翻の1,600点になります。これらの違いから最大値を引き出してくるため，すべての面子分け，すべてのアガリ牌の位置について点を数え，最大値となったときのみ点を表示させ，役の名前を書かせることにしました。

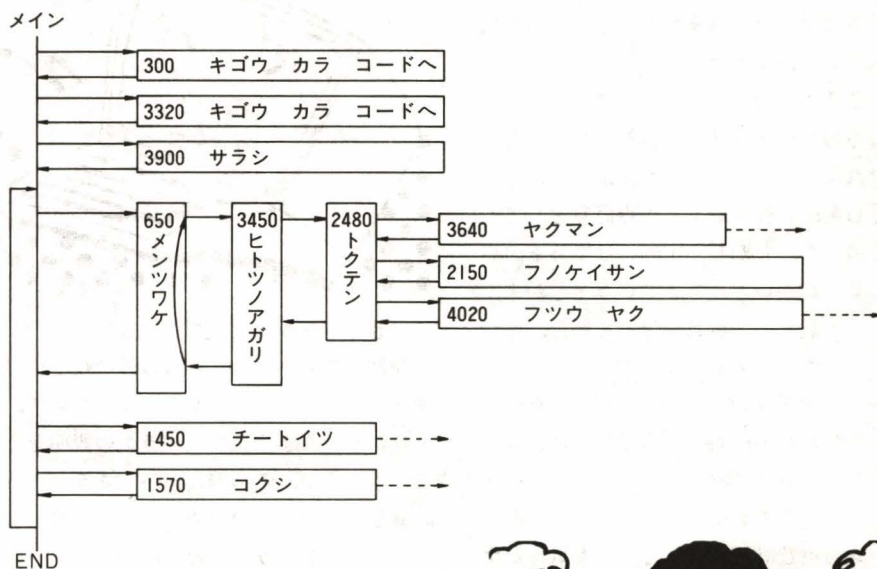



表1　サブルーチン 構造

では，サブルーチンの構造を見てもらいましょう（表１）。図が少し複雑になりましたが，メインの左側で戻る矢印があるのは，先ほど触れたように，ひととおりの和り方についてすべて計算したうえで最大値を求め，もう一度戻って，再度その最大の得点を見つけたところでディスプレイする，という形を示しているのです。

また，右端に点線で矢印があるのは，さらに小さなサブルーチンに向かうことを示しています。それでは，ひとつひとつのサブルーチンを，流れにそって説明していきます。

・300～　アガったときの牌14枚を記号（1P，8M……など）で入力し，それをコード番号にして配列Hに入れる。

H(I（(I＝1～14)

・3320～　得点計算に必要な情報（ドラ・場風，リーチの有無など）を入力したとき，それをメモリに格納する。

・3900～　喰いを入れたときはその情報を入力し，メモリに数の形で格納する。

・650～　普通の形のアガリ形をすべてのやり方で次々に分け，それを配列Bに入れる。

B(I)（I＝1～14）

・3450～　アガリ形ひとつの中で，アガリ

表2

B(I)	面子分けした手牌
F1	役の翻数
F2	役満の数
F3	符
F4	得点
H(I)	面子分けしない手牌
K(I)	ひとつひとつの面子がクイか手の内か(手の内0, 喰い1) I=5(K(5)=0)は手牌全体が面前かどうか(面前0, あとは1)
K6	アガリ牌の種類
K7	アガリ牌の手牌中での位置
K8	ツモアガリは0　ロンは1
K9	リーチは1　なしは0
L(I)	I=1～43　それぞれの枚数 I=47　老頭牌の数 I=49　字牌の数 I=51, 53, 55　万・筒・索子のそれぞれの枚数
M	アガリの形があれば>0
P(I)	喰いの情報 I=1～12　喰い牌 I=13～16　喰いの種類 ポンなら1, チーなら10, アンカンが100, ミンカンは1000
X7	得点の最大値
Y(I)	I=1～4　それぞれの面子が刻子だと1 順子だと0 I=　Y(5)=Y(1)+Y(2)+Y(3)+Y(4)
Y5	ドラ
Y6	場風
Y7	自分の風
Z$(I)	手牌　I=15　役満の名 I=16　役の名

牌の位置が2つ以上ある場合，そのすべてについてのループを行なう。

- ・2480～　ヤクマン，フノケイサン，フツヤクの順で役，符を調べ，親・子，ツモ・ロンによって得点計算する。
- ・3640～　ヤクマンであるかどうかを判断し，ヤクマンならその役の名前をZ＄(15)に入れる。
- ・2150～　そのアガリ手の符を計算する。それをF3にメモリする。
- ・4020～　フツウの役を数えてF1に記憶し，役の名はZ＄(16)に入れる。
- ・1450～　チートイツであるかどうかを調べ，そうであれば役判断に進む。
- ・1570～　コクシであるかどうかを調べ，そうであれば得点を表示する。

では次に，変数表を見てもらいましょう(表2)。役を判断するのに特に重要なのは，配列K，Y，L，Pです。この役判断の部分はかなり苦労した部分ですが，これでも配列の値を使って，短かくなったほうなのです。

4 実行と経過

次に，実際走らせてみてどういうことが行なわれるのかを，図2の例で順を追って見てみることにします。

まず，RUNすると，手牌の内容を聞いてきます。表3の記号のほうで14個のデータを入力してください。次にアガリ牌をやはり記号で，ツモ・ロン・リーチを0，1で，また，ドラ・場風・自風を記号で入れます。次に，この場合面前ですから，0を入力します。これで計算が始まります。まずHにはI=1～14まで，3，4，5，13，14，15，22，22，23，23，24，24，25，25と入り，それを面子分けでまずB(I)=3，4，5，13，14，15，23，24，25，23，24，25，22，22と雀頭を最後にして並べ換えます。そして，これがフツウヤクのルーチンに行くと4430行でサンショク，4710行でツモ，4720行でタンヤオにひっかかり，2480～のルーチンで得点計算をして，F4に合計点を入れます。

さて，面子分けに戻ったら，今度はB(I)=3，4，5，13，14，15，22，23，24，22，23，24，25，25と別の面子について得点を計算します。そして先程のF4と比べ，大きいほうの点をX7に入れておきます。他の形がないことを確認したら，もう一度最初の面子分けに戻り，今度は計算した点がX7に等しければ，その形での役と点を表示し，メインルーチンに戻って終わりです。

今までいろいろと出ているゲームの得点計算法は，この最大値をとるところがうまくいっていなかったため，まちがった点数を出すものが多かったのです。今回ももっとも苦心した点がこの部分でした。

さて，ここまでくると，このプログラムにない役でもヤクマンやフツウヤクのルーチンを伸ばして，いくらでも付け加えられることがおわかりいただけたと思います。しかしその際にはK，Y，Pの配列の値に気を払って，条件式にミスのないように作らなければなりません。それを考えるのが結構むずかしく，反面おもしろいですので，麻雀とBASICの両方にお強い方はチャレンジしてみてください。

表3

実際の牌	記号	コード番号
万子	1M～9M	1～9
筒子	1P～9P	11～19
索子	1S～9S	21～29
東	T	31
南	N	33
西	S	35
北	P	37
白	HK	39
発	HT	41
中	CH	43

```
10 'ﾄｸﾃﾝ ｹｲｻﾝ
20 DEFINTA-E,G-W,Y
30 DIM E(9),G(12),H(14),L(55),R(3),S(6),Z$(29),K(5),Y(8),B(14),P(20)
40 GOSUB300
50 GOSUB 580
60 INPUT"ｱｶﾞﾘﾊｲ ﾊ";C$
70 GOSUB3320:K6=F9
80 INPUT"ﾂﾓ(0),ﾛﾝ(1)";K8
90 INPUT"ﾘｰﾁ(1)";K9
100 INPUT"ﾄﾞﾗ";C$
110 GOSUB3320:Y5=F9
120 INPUT"ﾊﾞｶｾﾞ";C$
```

```
130 GOSUB3320:Y6=F9
140 INPUT"ｼﾞｶｾﾞ";C$
150 GOSUB3320:Y7=F9
160 FORI=1TO14:L(H(I))=L(H(I))+1:NEXTI
170 L(47)=L(1)+L(9)+L(11)+L(19)+L(21)+L(29)
180 FORI=1TO9:L(51)=L(51)+L(I):L(53)=L(53)+L(10+I):L(55)=L(55)+L(20+I):NEXTI
190 L(49)=14-L(51)-L(53)-L(55)
200 INPUT "ﾒﾝｾﾞﾝ ﾅﾗ 0 ﾁｶﾞｳ ﾄ 1";F9:IFF9 GOSUB3900
210 GOSUB650
220 IFP(13)<>0THEN240
230 GOSUB1450:GOSUB1570
240 IFM<>0THEN260
250 PRINT"ﾁｮﾝﾎﾞ":GOTO290
260 Y8=X7:IFX9<>0THEN280
270 X9=X9+1:GOTO210
280 PRINT"ｺﾞｳｹｲ ﾄｸﾃﾝﾊ";Y8;"ﾃﾝ"
290 END
300 'ｷｺﾞｳ ｶﾗ ｺｰﾄﾞ ﾍ
310 PRINT"ｱｶﾞﾘ ﾉ ｶﾀﾁ ｦ ｷｺﾞｳ ﾃﾞ ｲﾚﾃ ｸﾀﾞｻｲ"
320 FORI=1TO14:INPUTZ$(I):NEXTI
330 FORI=1TO14
340 IFZ$(I)="T"THENH(I)=31:GOTO450
350 IFZ$(I)="N"THENH(I)=33:GOTO450
360 IFZ$(I)="S"THENH(I)=35:GOTO450
370 IFZ$(I)="P"THENH(I)=37:GOTO450
380 IFZ$(I)="HK"THENH(I)=39:GOTO450
390 IFZ$(I)="HT"THENH(I)=41:GOTO450
400 IFZ$(I)="CH"THENH(I)=43:GOTO450
410 IFRIGHT$(Z$(I),1)="M"THENH(I)=VAL(LEFT$(Z$(I),1)):GOTO450
420 IFRIGHT$(Z$(I),1)="P"THENH(I)=VAL(LEFT$(Z$(I),1))+10:GOTO450
430 IFRIGHT$(Z$(I),1)="S"THENH(I)=VAL(LEFT$(Z$(I),1))+20:GOTO450
440 H(I)=45
450 NEXTI
460 RETURN
470 IFI<10THENJ$=STR$(I)+"M":GOTO570
480 IFI<20THENJ$=STR$(I-10)+"P":GOTO570
490 IFI<30THENJ$=STR$(I-20)+"S":GOTO570
500 IFI=31THENJ$="  T":GOTO570
510 IFI=33THENJ$="  N":GOTO570
520 IFI=35THENJ$="  S":GOTO570
530 IFI=37THENJ$="  P":GOTO570
540 IFI=39THENJ$=" HK":GOTO570
550 IFI=41THENJ$=" HT":GOTO570
560 IFI=43THENJ$=" CH":GOTO570
570 RETURN
580 C=1
590 IFC=14THEN640
600 J=C
610 FORI=C+1TO14:IFH(I)<H(J)THENJ=I
620 NEXTI
630 S=H(J):H(J)=H(C):H(C)=S:C=C+1:GOTO590
640 RETURN
650 'ﾒﾝﾂ ﾜｹ
660 FORI1=1TO13
670 IFI=1THEN690
680 IFH(I1)=H(I1-1)THEN780
690 IFH(I1)<>H(I1+1)THEN780
700 B(13)=H(I1):B(14)=H(I1)
710 I3=1
720 FORI2=1TO12
730 IFI2=I1THENI3=I3+2
740 G(I2)=H(I3):I3=I3+1
750 NEXTI2
760 GOSUB800
770 I1=I1+1
780 NEXTI1
790 RETURN
800 IFG(3)<>G(1)THEN840
810 FORI=1TO9:E(I)=G(I+3):NEXTI
820 Y(1)=1:B(1)=G(1):B(2)=G(2):B(3)=G(3)
830 GOSUB990
840 FORI=2TO12
850 IFG(I)<>G(1)+1THEN970
860 FORJ=I+1TO12
870 IFG(J)<>G(I)+1THEN960
880 IFI=2THEN900
890 FORX=1TOI-2:E(X)=G(X+1):NEXTX
900 IFJ=I+1THEN920
910 FORX=I-1TOJ-3:E(X)=G(X+2):NEXTX
920 IFJ=12THEN940
930 FORX=J-2TO9:E(X)=G(X+3):NEXTX
940 Y(1)=0:B(1)=G(1):B(2)=G(I):B(3)=G(J)
```

```
950 GOSUB990:J=12:I=12
960 NEXTJ
970 NEXTI
980 RETURN
990 IFE(3)<>E(1)THEN1030
1000 FORI=1TO6:S(I)=E(I+3):NEXTI
1010 Y(2)=1:B(4)=E(1):B(5)=E(2):B(6)=E(3)
1020 GOSUB1180
1030 FORI=2TO9
1040 IFE(I)<>E(1)+1THEN1160
1050 FORJ=I+1TO9
1060 IFE(J)<>E(I)+1THEN1150
1070 IFI=2THEN1090
1080 FORX=1TOI-2:S(X)=E(X+1):NEXTX
1090 IFJ=I+1THEN1110
1100 FORX=I-1TOJ-3:S(X)=E(X+2):NEXTX
1110 IFJ=9 THEN1130
1120 FORX=J-2TO6:S(X)=E(X+3):NEXTX
1130 Y(2)=0:B(4)=E(1):B(5)=E(I):B(6)=E(J)
1140 GOSUB1180:J=9 :I=9
1150 NEXTJ
1160 NEXTI
1170 RETURN
1180 IFS(3)<>S(1)THEN1220
1190 FORI=1TO3:R(I)=S(I+3):NEXTI
1200 Y(3)=1:B(7)=S(1):B(8)=S(2):B(9)=S(3)
1210 GOSUB1370
1220 FORI=2TO6
1230 IFS(I)<>S(1)+1THEN1350
1240 FORJ=I+1TO6
1250 IFS(J)<>S(I)+1THEN1340
1260 IFI=2THEN1280
1270 FORX=1TOI-2:R(X)=S(X+1):NEXTX
1280 IFJ=I+1THEN1300
1290 FORX=I-1TOJ-3:R(X)=S(X+2):NEXTX
1300 IFJ=6 THEN1320
1310 FORX=J-2TO3:R(X)=S(X+3):NEXTX
1320 Y(3)=0:B(7)=S(1):B(8)=S(I):B(9)=S(J)
1330 GOSUB1370:J=6 :I=6
1340 NEXTJ
1350 NEXTI
1360 RETURN
1370 IFR(3)=R(1)THEN1410
1380 IFR(2)<>R(1)+1THEN1440
1390 IFR(3)<>R(2)+1THEN1440
1400 Y(4)=0:GOTO1420
1410 Y(4)=1
1420 M=M+1:B(10)=R(1):B(11)=R(2):B(12)=R(3)
1430 GOSUB3450
1440 RETURN
1450 'ﾁｰﾄｲﾂ
1460 GOSUB580
1470 J=0
1480 FORI=1TO7
1490 IFH(2*I)=H(2*I-1)THENJ=J+1
1500 NEXTI
1510 FORI=1TO6
1520 IFH(2*I)=H(2*I+1)THENJ=J-1
1530 NEXTI
1540 IFJ<7THEN1560
1550 M=M+1:GOSUB3670
1560 RETURN
1570 'ｺｸｼ
1580 GOSUB580
1590 FORI=1TO13
1600 IFH(I+1)=H(I)THENX=I:N=0
1610 NEXTI
1620 IFN>0THEN1740
1630 FORJ=XTO13:H(J)=H(J+1):NEXTJ
1640 FORI=1TO3
1650 IFH(2*I-1)=10*I-9THENN=N+1
1660 IFH(2*I)=10*I-1THENN=N+1
1670 NEXTI
1680 FORI=7TO13
1690 IFH(I)=2*I+17THENN=N+1
1700 NEXTI
1710 FORJ=14TOX+1STEP-1:H(J)=H(J-1):NEXTJ
1720 IFN=13THENM=M+1ELSE1740
1730 GOSUB3770
1740 RETURN
1750 'ﾒﾝﾂ ﾋｮｳｼﾞ
1760 FORC1=1TO4:FORC2=1TO3:I=B(3*C1-3+C2):GOSUB470:PRINTJ$;:NEXTC2:PRINT"":NEXTC
```

```
1770 I=B(13):GOSUB470:PRINTJ$;J$:RETURN
1780 'ﾔｸﾏﾝ
1790 IFB(4)=39THENF2=F2+1:Z$(15)=Z$(15)+" ﾀﾞｲｻﾝｹﾞﾝ"
1800 N=0
1810 FORX1=1TO4
1820 IFP(12+X1)>90THENN=N+1
1830 NEXTX1
1840 IFN=4THENF2=F2+1:Z$(15)=Z$(15)+" ｽｰｶﾝﾂ"
1850 IFB(1)<>31THEN1870
1860 IFB(10)=37THENF2=F2+2:Z$(15)=Z$(15)+" ﾀﾞｲｽｰｼ":GOTO1880
1870 IFL(31)+L(33)+L(35)+L(37)>10THENF2=F2+1:Z$(15)=Z$(15)+" ｼｮｳｽｰｼ"
1880 IFK(5)>0THEN1950
1890 IFY(1)+Y(2)+Y(3)+Y(4)<4THEN1950
1900 IFK8=0THEN1920
1910 IFK7<13THEN1950
1920 F2=F2+1:Z$(15)=Z$(15)+" ｽｰｱﾝｺ"
1930 IFK7<13THEN1950
1940 F2=F2+1:Z$(15)=Z$(15)+" ﾀﾝｷ"
1950 IFL(47)>13THENF2=F2+1:Z$(15)=Z$(15)+" ﾁﾝﾛｰﾄｰ":GOTO2140
1960 IFH(14)>30THEN2110
1970 IFK(5)>0THEN2110
1980 IFP(13)+P(14)+P(15)+P(16)>50THEN2110
1990 IFINT(H(1)/10)<>INT(H(14)/10)THEN2110
2000 IFL(1+10*INT(H(1)/10))*L(9+10*INT(H(1)/10))<9THEN2110
2010 N=0
2020 FORX1=1TO9
2030 IFL(X1+10*INT(H(1)/10))=0THENN=14
2040 NEXTX1
2050 IFN=0THENF2=F2+1:Z$(15)=Z$(15)+" ﾁｭｰﾚﾝ"
2060 IFINT(K6MOD10)=1THEN2100
2070 IFINT(K6MOD10)=9THEN2100
2080 IFL(K6)=1THEN2140
2090 F2=F2+1:Z$(15)=Z$(15)+" 9ﾒﾝ":GOTO2140
2100 IFL(K6)=4THENF2=F2+1:Z$(15)=Z$(15)+" 9ﾒﾝ":GOTO2140
2110 IFL(22)+L(23)+L(24)+L(26)+L(28)+L(41)>13THENF2=F2+1:Z$(15)=Z$(15)+" ﾘｭｰｲｰｿ"
2120 IFH(1)<30THEN2140
2130 IFH(13)>30THENF2=F2+1:Z$(15)=Z$(15)+" ﾂｰｲｰｿ"
2140 RETURN
2150 'ﾌ ﾉ ｹｲｻﾝ
2160 F3=20
2170 'ｺｰﾂ
2180 FORX1=1TO4:F4=0
2190 IFY(X1)=0THEN2320
2200 F4=F4+2
2210 IFP(12+X1)>90THENF4=F4*4
2220 IFP(3*X1)>30THENF4=F4*2:GOTO2250
2230 IFINT(P(3*X1)MOD10)=9THENF4=F4*2
2240 IFINT(P(3*X1)MOD10)=1THENF4=F4*2
2250 IFK(X1)=1THEN2310
2260 IFK7<3*X1-2THEN2300
2270 IFK7>3*X1THEN2300
2280 IFK8=0THEN2300
2290 GOTO2310
2300 F4=F4*2
2310 F3=F3+F4
2320 NEXTX1
2330 'ﾄｲﾂ
2340 IFB(14)>38THENF3=F3+2
2350 IFB(14)=Y6THENF3=F3+2
2360 IFB(14)=Y7THENF3=F3+2
2370 'ﾏﾁ
2380 IFK8=0THENF3=F3+2
2390 IFK7>12THENF3=F3+2:GOTO2440
2400 X1=INT(K7MOD3):X2=INT(K6MOD10)
2410 YY=Y(K7¥3+1):IFX1=2ANDYY=0THENF3=F3+2
2420 IFX1=0ANDX2=3THENF3=F3+2
2430 IFX1=1ANDX2=7THENF3=F3+2
2440 IFK(5)=0ANDK8=1THENF3=F3+10
2450 F4=F3
2460 F3=10*INT((F3-1)/10)+10
2470 RETURN
2480 'ﾄｸﾃﾝ
2490 'ﾔｸﾏﾝ
2500                IFK8=0THEN2530
2510 IFY7=31THENF5=F2*48000!:GOTO2550
2520 F5=F2*32000:GOTO2550
2530 IFY7=31THENF6=F2*16000:GOTO2550
2540 F6=F2*8000:F7=F2*16000
2550 IFY7=31THENF4=F5+3*F6:GOTO2570
2560 F4=F5+2*F6+F7
2570 IFX9=0THEN2600
2580 IFF4<>Y8THEN2600
```

```
2590 GOSUB1750:GOSUB3030
2600 RETURN
2610 'ﾏﾝｶﾞﾝ
2620 IFF1>9THENF8=6000:GOTO2680
2630 IFF1>7THENF8=4000:GOTO2680
2640 IFF1>5THENF8=3000:GOTO2680
2650 IFF1>4THENF8=2000:GOTO2680
2660 IFF1=4ANDF3>30THENF8=2000:GOTO2680
2670 IFF1=3ANDF3>70THENF8*2000
2680 IFK8=0THEN2710
2690 IFY7=31THENF5=6*F8:GOTO2890
2700 F5=4*F8:GOTO2890
2710 IFY7=31THENF6=2*F8:GOTO2890
2720 F6=F8:F7=2*F8:GOTO2890
2730 'ﾏﾝｶﾞﾝ ﾐﾏﾝ
2740 IFK8=0THEN2810
2750 IFY7=31THEN2780
2760 F5=100*INT((16*2↑F1*F3-5)/100)+100
2770 IFF5>8000THENF5=8000:GOTO2890
2780 F5=100*INT((24*2↑F1*F3-5)/100)+100
2790 IFF5>12000THENF5=12000
2800 GOTO2890
2810 IFY7=31THEN2870
2820 F6=100*INT((4*2↑F1*F3-5)/100)+100
2830 F7=100*INT((8*2↑F1*F3-5)/100)+100
2840 IFF6>2000THENF6=2000
2850 IFF7>4000THENF7=4000
2860 GOTO2890
2870 F6=100*INT((8*2↑F1*F3-5)/100)+100
2880 IFF6>4000THENF6=4000
2890 RETURN
2900 'ﾄｸﾃﾝ ﾃﾞｨｽﾌﾟﾚｲ
2910 X9=2:PRINTZ$(16)
2920 IFK8=0THEN2960
2930 IFY7=31THEN2950
2940 PRINT"ｺ ﾉ ﾛﾝ ｱｶﾞﾘ";F3;"ﾌ";F1;"ﾌｧﾝ";F5;"ﾃﾝ":GOTO3020
2950 PRINT"ｵﾔ ﾉ ﾛﾝ ｱｶﾞﾘ";F3;"ﾌ";F1;"ﾌｧﾝ";F5;"ﾃﾝ":GOTO3020
2960 IFY7=31THEN3000
2970 PRINT"ｺ ﾉ ﾂﾓ ｱｶﾞﾘ";F3;"ﾌ";F1;"ﾌｧﾝ"
2980 PRINT"ｵﾔ ｶﾗ";F7;"ﾃﾝ"
2990 PRINT"ｺ ｶﾗ";F6;"ﾃﾝ":GOTO3020
3000 PRINT"ｵﾔ ﾉ ﾂﾓ ｱｶﾞﾘ";F3;"ﾌ";F1;"ﾌｧﾝ"
3010 PRINT"ｺ ｶﾗ";F6;"ﾃﾝ ｽﾞﾂ"
3020 RETURN
3030 'ﾔｸﾏﾝ ﾃﾞｨｽﾌﾟﾚｲ
3040 PRINTZ$(15):X9=2
3050 IFK8=0THEN3080
3060 IFY7=31THENPRINT"ｵﾔ ﾉ ﾛﾝ ｱｶﾞﾘ";:ELSEPRINT"ｺ ﾉ ﾛﾝ ｱｶﾞﾘ";
3070 PRINTF5;"ﾃﾝ":GOTO3100
3080 IFY7=31THENPRINT"ｵﾔ ﾉ ﾂﾓ ｱｶﾞﾘ":PRINT"ｺ ｶﾗ";F6;"ﾃﾝ ｽﾞﾂ":GOTO3100
3090 PRINT"ｺ ﾉ ﾂﾓ ｱｶﾞﾘ":"ｵﾔ ｶﾗ";F7;"ﾃﾝ":PRINT"ｺ ｶﾗ";F6;"ﾃﾝ"
3100 RETURN
3110 'ﾄｸﾃﾝ
3120 GOSUB1780
3130 IFF2>0THENGOSUB2490:GOTO3200
3140 GOSUB2150:GOSUB4020:GOSUB2610
3150 IFY7=31THENF4=F5+3*F6:GOTO3170
3160 F4=F5+2*F6+F7
3170 IFX9=0THEN3200
3180 IFF4<>Y8THEN3200
3190 GOSUB1750:GOSUB2900
3200 RETURN
3210 'ﾁｰﾄｲﾂ ﾉ ﾔｸ
3220 F2=0
3230 GOSUB2120
3240 IFF2=0THEN3260
3250 GOSUB2490:GOTO3310
3260 GOSUB4700:GOSUB2610
3270 IFY7=31THENF4=F5+3*F6:GOTO3290
3280 F4=F5+2*F6+F7
3290 IFX9=0THEN3310
3300 IFF4=Y8THENGOSUB1750:GOSUB2900
3310 RETURN
3320 'ｷｺﾞｳ ｶﾗ ｺｰﾄﾞ
3330 IFC$="T"THENF9=31:GOTO3440
3340 IFC$="N"THENF9=33:GOTO3440
3350 IFC$="S"THENF9=35:GOTO3440
3360 IFC$="P"THENF9=37:GOTO3440
3370 IFC$="HK"THENF9=39:GOTO3440
3380 IFC$="HT"THENF9=41:GOTO3440
3390 IFC$="CH"THENF9=43:GOTO3440
3400 IFRIGHT$(C$,1)="M"THENF9=VAL(LEFT$(C$,1)):GOTO3440
```

```
3410 IFRIGHT$(C$,1)="P"THENF9=VAL(LEFT$(C$,1))+10:GOTO3440
3420 IFRIGHT$(C$,1)="S"THENF9=VAL(LEFT$(C$,1))+20:GOTO3440
3430 F9=45
3440 RETURN
3450 'ﾋﾄﾂ ﾉ ｱｶﾞﾘ
3460 FORX1=1TO4:K(X1)=0:NEXTX1
3470 FORX2=1TO4
3480 IFP(3*X2-2)=0THENX2=4:X1=0:GOTO3560ELSEX1=4
3490 FORX3=1TO4
3500 IF(P(3*X2-2)-B(3*X3-2))↑2+(P(3*X2-1)-B(3*X3-1))↑2=0THENX1=0:X4=X3:X3=4
3510 NEXTX3
3520 IFX1>0THENM=M-1:X2=4:GOTO3560
3530 P(16+X4)=X2
3540 IFP(12+X2)=100THEN3560
3550 K(X4)=1
3560 NEXTX2
3570 IFX1>0THEN3660
3580 K(5)=K(1)+K(2)+K(3)+K(4)
3590 FORX8=1TO13
3600 IFX9=2THENX8=13:GOTO3650
3610 IFB(X8)<>K6THEN3630
3620 K7=X8:F1=0:F2=0:F3=0:F4=0:F5=0:F6=0:F7=0:F8=0:GOSUB3110
3630 IFF4>=X7THENX7=F4
3640 Z$(16)="":Z$(15)=""
3650 NEXTX8
3660 RETURN
3670 ' ﾁｰﾄｲﾂ ﾄｸﾃﾝ
3680 FORX1=1TO14:B(X1)=H(X1):NEXTX1
3690 F1=0:F2=0:F3=0:F5=0:F6=0:F7=0:PRINT
3700 Z$(15)=Z$(15)+" ﾁｰﾄｲﾂ":Z$(16)=Z$(16)+" ﾁｰﾄｲﾂ"
3710 F1=2:F3=25
3720 GOSUB3210
3730 IFY7=31THENF4=F5+3*F6:GOTO3750
3740 F4=F5+2*F6+F7
3750 IFF4>=X7THENX7=F4
3760 Z$(16)="":Z$(15)="":RETURN
3770 'ｺｸｼ ﾉ ﾄｸﾃﾝ
3780 F1=0:F2=1:F3=0:F5=0:F6=0:F7=0:X2=0
3790 FORX1=1TO14:B(X1)=H(X1)
3800 IFB(X1)=K6THENX2=X2+1
3810 NEXTX1
3820 PRINT
3830 Z$(15)=Z$(15)+" ｺｸｼ"
3840 IFX2<>1THENF2=F2+1:Z$(15)=Z$(15)+" 13ﾒﾝﾏﾁ"
3850 GOSUB2490
3860 IFY7=31THENF4=F5+3*F6:GOTO3880
3870 F4=F5+2*F6+F7
3880 IFF4>=X7THENX7=F4
3890 Z$(15)="":Z$(16)="":RETURN
3900 'ｻﾗｼ
3910 FORX1=1TO20:P(X1)=0:NEXTX1
3920 FORX1=1TO4
3930 INPUT"ｸｲ ﾉ ｶﾀﾁ ﾊ --PN,TI,AK,MK";Z$(17)
3940 INPUT"ｸｲ ﾊｲ ﾊ";C$
3950 GOSUB3320
3960 IFZ$(17)="PN"THENP(3*X1-2)=F9:P(3*X1-1)=F9:P(3*X1)=F9:Y(4+X1)=1:P(12+X1)=1:
GOTO4000
3970 IFZ$(17)="TI"THENP(3*X1-2)=F9:P(3*X1-1)=F9+1:P(3*X1)=F9+2:P(12+X1)=10:GOTO4
000
3980 IFZ$(17)="AK"THENP(3*X1-2)=F9:P(3*X1-1)=F9:P(3*X1)=F9:Y(4+X1)=1:P(12+X1)=10
0:GOTO4000
3990 IFZ$(17)="MK"THENP(3*X1-2)=F9:P(3*X1-1)=F9:P(3*X1)=F9:Y(4+X1)=1:P(12+X1)=10
00
4000 NEXTX1
4010 RETURN
4020 'ﾌﾂｳ ﾔｸ
4030 Y(5)=Y(1)+Y(2)+Y(3)+Y(4)
4040 IFK(5)<>0THEN4110
4050 IFK8=0ANDF4=22THENF3=20:F1=F1+1:Z$(16)=Z$(16)+" ﾋﾟﾝﾌ"
4060 IFK8=1ANDF4=30THENF1=F1+1:Z$(16)=Z$(16)+" ﾋﾟﾝﾌ"
4070 IFY(5)=0ANDB(1)=B(4)ANDB(7)=B(10)THENF1=F1+3:Z$(16)=Z$(16)+" ﾘｬﾝﾍﾟｰｺ":GOTO4
160
4080 FORX1=1TO3
4090 IFB(3*X1-2)=B(3*X1+1)ANDY(X1)+Y(X1+1)=0THENF1=F1+1:Z$(16)=Z$(16)+" ｲｰﾍﾟｰｺ":
X1=3
4100 NEXTX1
4110 IFL(Y6)>2THENF1=F1+1:Z$(16)=Z$(16)+" ﾔｸﾊｲ"
4120 IFL(Y7)>2THENF1=F1+1:Z$(16)=Z$(16)+" ﾔｸﾊｲ"
4130 FORX1=39TO43STEP2
4140 IFL(X1)>2THENF1=F1+1:Z$(16)=Z$(16)+" ﾔｸﾊｲ"
4150 NEXTX1
4160 FORX1=1TO4:X4=0:FORX2=1TO3:X3=B(3*X1-3+X2)
```

```
4170 GOSUB4280:NEXTX2
4180 IFX4=0THENX1=4:X2=0ELSEX2=4
4190 NEXTX1
4200 IFX2=0THEN4310
4210 X4=0:X3=B(13):GOSUB4280
4220 IFX4=0THEN4310
4230 IFY(5)=4THEN4310
4240 IFL(49)>0THENF1=F1+1:Z$(16)=Z$(16)+" ﾎﾝﾁｬﾝﾀ":GOTO4260
4250 F1=F1+2:Z$(16)=Z$(16)+" ｼﾞｭﾝﾁｬﾝ"
4260 IFK(5)=0THENF1=F1+1
4270 GOTO4310
4280 REMﾔｵﾁｭｰ
4290 IFX3>30ORINT(X3MOD10)=9ORINT(X3MOD10)=1THENX4=X4+1
4300 RETURN
4310 RESTORE4390
4320 FORX4=1TO4
4330 READX1,X2,X3
4340 IFINT(B(X1)MOD10)=1ANDB(X2)=B(X1)+3ANDB(X3)=B(X2)+3ANDY((X1+2)/3)+Y((X2+2)/
3)+Y((X3+2)/3)=0THENX4=4:X1=0
4350 NEXTX4
4360 IFX1>0THEN4400
4370 F1=F1+1:Z$(16)=Z$(16)+" ｲｯﾂｰ"
4380 IFK(5)=0THENF1=F1+1
4390 DATA1,4,7,1,4,10,1,7,10,4,7,10
4400 RESTORE4390
4410 FORX4=1TO4
4420 READX1,X2,X3
4430 IFB(X2)=B(X1)+10ANDB(X3)=B(X2)+10ANDY((X1+2)/3)+Y((X2+2)/3)+Y((X3+2)/3)=0TH
ENF1=F1+1:Z$(16)=Z$(16)+" ｻﾝｼｮｸ":X4=4ELSE4450
4440 IFK(5)=0THENF1=F1+1
4450 NEXTX4
4460 IFY(5)=4THENF1=F1+2:Z$(16)=Z$(16)+" ﾄｲﾄｲ"
4470 X2=0
4480 FORX1=1TO4
4490 IFP(12+X1)>99THENX2=X2+1
4500 NEXTX1
4510 IFX2=3THENF1=F1+2:Z$(16)=Z$(16)+" ｻﾝｶﾝﾂ"
4520 RESTORE4390
4530 FORX4=1TO4
4540 READX1,X2,X3
4550 IFY((X1+2)/3)+Y((X2+2)/3)+Y((X3+2)/3)<>3ORK((X1+2)/3)+K((X2+2)/3)+K((X3+2)/
3)<>0THEN4620
4560 IFK8=0THEN4610
4570 FORX5=1TO3
4580 X6=K7+3*X2-X5
4590 IFX6=3*X1ORX6=3*X2ORX6=3*X3THENX6=0:X5=3
4600 NEXTX5
4610 F1=F1+2:Z$(16)=Z$(16)+" ｻﾝｱﾝｺ":X4=4
4620 NEXTX4
4630 IFL(39)+L(41)+L(43)>7THENF1=F1+2:Z$(16)=Z$(16)+" ｼｮｰｻﾝｹﾞﾝ":GOTO4700
4640 RESTORE4390
4650 FORX4=1TO4
4660 READX1,X2,X3
4670 IFY((X1+2)/3)+Y((X2+2)/3)+Y((X3+2)/3)<>3THEN4690
4680 IFINT(B(X1)MOD10)=INT(B(X2)MOD10)ANDINT(B(X2)MOD10)=INT(B(X3)MOD10)THENF1=F
1+2:Z$(16)=Z$(16)+" ｻﾝｼｮｸﾄﾞｳﾎﾟﾝ"
4690 NEXT X4
4700 'ﾌﾂｳ ﾔｸ(ﾁｰﾄｲﾂ)
4710 IFK(5)+K8=0THENF1=F1+1:Z$(16)=Z$(16)+" ﾂﾓ"
4720 IFL(47)+L(49)=0THENF1=F1+1:Z$(16)=Z$(16)+" ﾀﾝﾔｵ"
4730 IFK9=1THENF1=F1+1:Z$(16)=Z$(16)+" ﾘｰﾁ"
4740 IFL(Y5)>0THENF1=F1+L(Y5):Z$(16)=Z$(16)+" ﾄﾞﾗ"+STR$(L(Y5))
4750 IFL(47)+L(49)>13THENF1=F1+2:Z$(16)=Z$(16)+" ﾎﾝﾛｰﾄｰ"
4760 FORX1=1TO3
4770 IFL(49+2*X1)+L(49)>13ANDL(49)>0THENF1=F1+2:Z$(16)=Z$(16)+" ﾎﾝｲﾁ"ELSE4800
4780 IFK(5)=0THENF1=F1+1
4790 X1=3
4800 NEXTX1
4810 FORX1=1TO3
4820 IFL(49+2*X1)>13THENF1=F1+5:Z$(16)=Z$(16)+" ﾁﾝｲﾁ"ELSE4840
4830 X1=3:IFK(5)=0THENF1=F1+1
4840 NEXTX1
4850 RETURN
```

PCG700(MZ-700)

気分はもうゲームセンター

誰でも作れるフルグラフィック

景山 利之

MZ-700は，その実行速度やBASICの強力さにおいては数十万もするようなパーソナルコンピュータと比較しても決してひけはとりませんが，細かいドットを打つことができないため，ゲームを作るときなどは苦労させられます。

私のようにもっぱら自作のハードウェアのテスト用に使う場合には，英数字と若干の記号があれば事足りるのですが，ゲームの場合には，やはりそれだけではさみしいものです。

仕方なく内蔵されたキャラクタセットを組み合わせるわけですが，ヘッドオンをやってもパックマンをやっても，相手がUFOモドキやニコチャンマークでは今ひとつ楽しくありません。

「何とかならんか！」というわけで考えられたのが，ここで紹介するPCG-700です。

PCG-700は，MZの持っている，カナやニコチャンマークなどのキャラクタの形を自由に書き換えられるようにしてしまいます。

ニコチャンマークやUFOモドキを自分で定義したキャラクタにすれば，気分はもうゲームセンター。本物そっくりのインベーダーやパックマンが動きまわる様子はすばらしいもので，思わず「これ，MZ-700，だよねェ？」と口にでてしまいました。

「面白そうだけど，パターンを定義するのが難しいんじゃない」と思う人も心配御無用。PCG-700にはパターンを作るためのソフトウェアとして「PCG AID」が付いていますから，非常に簡単にオリジナルパターンをつくることができます。このソフトもよくできていて，それだけでも数千円の価値があるんじゃないかと思いたくなります。

●PCG-700を使ったフルグラフィック

編集部に到着したPCG-700をさっそく使ってみました。取扱い説明書にあった，BASICによるパターン定義をやっているうちに，ふと「フルグラフィックができるんじゃないか」と思い，さっそくやってみましたが，どうもおかしいのです。書き込むつもりのないところに，データが書き込まれることがあります。1時間近く調べてわかったのは「PCGのキャラクタが表示されているときに，キャラクタの定義をしてはいけない」ということでした。

まあ，PCGは普通キャラクタを定義したあとに，そのキャラクタを使いますから，あまり気にしなくても良いのですが，一応，覚えておいたほうがよいかもしれません。

ですから，スイッチは，ONの状態にしておいてください。

話が少しそれてしまいましたが，上記のことに気を付けて，プログラムを組んだところ，具合よく動作し，MZ-700で128×128ドットのフルグラフィックが行なえるようになりました。「何だ，たった128×128か」なんて言わないでください。ドットの細かさは，MZのキャラクタの1ドットと同じですから，なかなかいけます。

●プログラムの説明

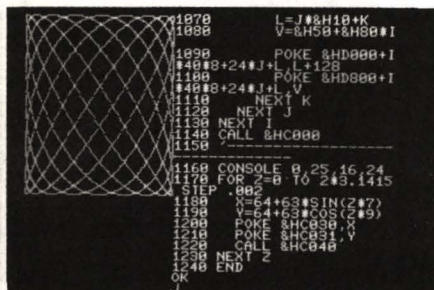
このプログラムでは，ワークエリアとして，C400～C7FFの2Kバイトを使用します。ここには，PCGのRAMと同じデータを入れています。これはPCG側のRAMの内容を読み出すことができないためです。

プログラム自体は，このワークエリアとPCGのRAMをクリアする部分(C000番地から）と，C030番地をX座標，C031番地をY座標とみなして，指定されたところに点を打つ部分(C040番地）からなります。

具体的な使い方はBASIC部を見てください。

1040行～1130行でキャラクタを並べ，CALL 8H000でクリアします。

続いて，計算で得たX座標を&HC030に，Y座標を&HC031にPOKEして，CALL &HC040でドットを打ちます。

座標は，左上の隅が原点(0, 0)になっていますので，数学の教科書でみなれている座標系にしたい人は，128-Yとしてください。

```
C000  DD 21 10 E0 21 00 C4 01    C058  4F DD 5E 01 16 00 7B E6
C008  00 18 DD 36 02 08 36 00    C060  07 5F 09 19 DD 7E 00 E6
C010  DD 36 00 00 DD 71 01 DD    C068  07 ED 44 C6 07 47 0E 01
C018  70 02 DD 36 02 08 23 03    C070  FE 00 28 05 CB 21 05 20
C020  78 FE 20 20 E9 DD 36 02    C078  FB FD 21 00 C4 54 5D FD
C028  00 C9 00 00 00 00 00 00    C080  19 FD 7E 00 00 B1 FD 77
C030  00 00 00 00 00 00 00 00    C088  00 DD 21 10 E0 DD 36 02
C038  00 00 00 00 00 00 00 00    C090  08 DD 77 00 DD 75 01 7C
C040  DD 21 30 C0 DD 6E 01 26    C098  F6 18 DD 77 02 DD 36 02
C048  00 29 29 29 29 7D E6 80    C0A0  08 DD 36 02 00 C9 00 FF
C050  6F DD 7E 00 06 00 E6 FB    C0A8  00 FF 00 FF 00 FF 00 FF
```

```
1000 '----- PCG FULL GRAPHIC PROGRAM -------
1010 DEFINT A-Y
1020 CLEAR &HC000
1030 CLS
1040 FOR I=0 TO 1
1050   FOR J=0 TO 7
1060     FOR K=0 TO 15
1070       L=J*&H10+K
1080       V=&H50+&H80*I
1090       POKE &HD000+I*40*8+24*J+L,L+128
1100       POKE &HD800+I*40*8+24*J+L,V
1110     NEXT K
1120   NEXT J
1130 NEXT I
1140 CALL &HC000
1150 '--------------------------------
1160 CONSOLE 0,25,16,24
1170 FOR Z=0 TO 2*3.1415 STEP .01
1180   X=60+50*SIN(Z*5)
1190   Y=60+50*COS(Z*7)
1200   POKE &HC030,X
1210   POKE &HC031,Y
1220   CALL &HC040
1230 NEXT Z
1240 END
```

```
  PAGE    1
E STNR ADRS OBJECT     SOURCE STATEMENTS

     1                 ;
     2                 ; PCG DOT DSPLAY PROGRAM
     3                 ;
     4                 ; CLEAR PCG RAM
     5                 ;     &BUFF RAM
     6                 ;
     7 0000                    ORG  0C000H
     8 C000 DD2110E0           LD   IX,0E010H
     9 C004 2100C4             LD   HL,0C400H
    10 C007 010018             LD   BC,1800H
    11 C00A DD360208           LD   (IX+2),08H
    12 C00E 3600       LOOP:   LD   (HL),00H
    13 C010 DD360000           LD   (IX+0),00H
    14 C014 DD7101             LD   (IX+1),C
    15 C017 DD7002             LD   (IX+2),B
    16 C01A DD360208           LD   (IX+2),08H
    17 C01E 23                 INC  HL
    18 C01F 03                 INC  BC
    19 C020 78                 LD   A,B
    20 C021 FE20               CP   20H
    21 C023 20E9               JR   NZ,LOOP
    22 C025 DD360200           LD   (IX+2),00H
    23 C029 C9                 RET
    24                 ;
    25                 ;
    26                 ; DOT WRITE ROUTINE
    27                 ;
    28 C02A                    ORG  0C040H
    29                 ;
    30                 ; HL=INT(Y/8)*128
    31                 ;
    32 C040 DD2130C0           LD   IX,0C030H
    33 C044 DD6E01             LD   L,(IX+1)
    34 C047 2600               LD   H,00H
    35 C049 29                 ADD  HL,HL
    36 C04A 29                 ADD  HL,HL
    37 C04B 29                 ADD  HL,HL
    38 C04C 29                 ADD  HL,HL
    39 C04D 7D                 LD   A,L
    40 C04E E680               AND  80H
    41 C050 6F                 LD   L,A
    42                 ;
    43                 ; BC=8*INT(X/8)
    44                 ;
    45 C051 DD7E00             LD   A,(IX+0)
    46 C054 0600               LD   B,00H
    47 C056 E6F8               AND  0F8H
    48 C058 4F                 LD   C,A
    49                 ;
    50                 ; DE=(Y MOD 8)
    51                 ;
    52 C059 DD5E01             LD   E,(IX+1)
    53 C05C 1600               LD   D,00H
    54 C05E 7B                 LD   A,E
    55 C05F E607               AND  07H
    56 C061 5F                 LD   E,A
    57                 ;
    58                 ;
    59                 ; HL=HL+BC+DE
    60                 ;

  PAGE    2
E STNR ADRS OBJECT     SOURCE STATEMENTS

    61 C062 09                 ADD  HL,BC
    62 C063 19                 ADD  HL,DE
    63                 ;
    64                 ;
    65                 ; MAKE WRITE DATA
    66                 ;
    67                 ;
    68                 ; B=7-(X MOD 8)
    69                 ;
    70 C064 DD7E00             LD   A,(IX+0)
    71 C067 E607               AND  07H
    72 C069 ED44               NEG
    73 C06B C607               ADD  A,07H
    74 C06D 47                 LD   B,A
    75                 ;
    76                 ; MAKE 'OR BIT' PATTERN
    77                 ;
    78 C06E 0E01               LD   C,01H
    79 C070 FE00               CP   00H
    80 C072 2805               JR   Z,PSS1
    81 C074 CB21       LOOP2:  SLA  C
    82 C076 05                 DEC  B
    83 C077 20FB               JR   NZ,LOOP2
    84                 ;
    85                 ; MAKE DATA
    86                 ;
    87 C079 FD2100C4   PSS1:   LD   IY,0C400H
    88 C07D 54                 LD   D,H
    89 C07E 5D                 LD   E,L
    90 C07F FD19               ADD  IY,DE
    91 C081 FD7E00             LD   A,(IY+0)
    92 C084 00                 NOP
    93 C085 B1                 OR   C
    94 C086 FD7700             LD   (IY+0),A
    95                 ;
    96                 ; OUTPUT TO PCG
    97                 ;
    98 C089 DD2110E0           LD   IX,0E010H
    99 C08D DD360208           LD   (IX+2),08H
   100 C091 DD7700             LD   (IX+0),A
   101 C094 DD7501             LD   (IX+1),L
   102 C097 7C                 LD   A,H
   103 C098 F618               OR   18H
   104 C09A DD7702             LD   (IX+2),A
   105 C09D DD360208           LD   (IX+2),08H
   106 C0A1 DD360200           LD   (IX+2),0
   107 C0A5 C9                 RET
   108 C0A6                    END

ERRORS    0

SYMBOL ADRS  SYMBOL ADRS  SYMBOL ADRS

LOOP2  C074  LOOP   C00E  PSS1   C079
```

MZ-80K/C/1200/700

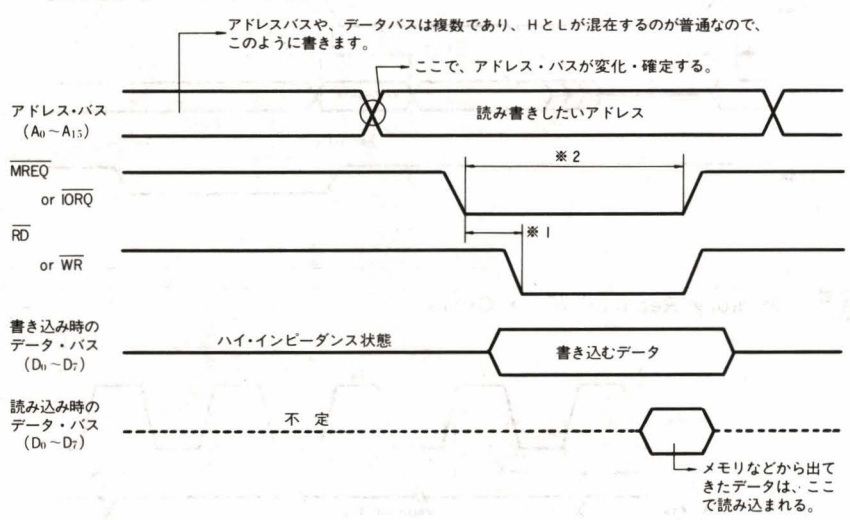
ハード入門 CPU Z-80の基本動作 II

早稲田大学宇宙航空研究会 棗野 雅彦

今月は、CPUのタイミングを見ていきましょう。

先月も述べましたが、マニュアルのタイミング図は，かなり複雑なものです。とりあえず，現在のところは図1位の，ごく大ざっぱなところがわかっていれば充分ですが，皆さんの技術レベルが上がったときのために，細かいタイミングも載せておきます。

図1 とりあえず、これだけは知っておきたい

※1：$\overline{\mathrm{WR}}$は、$\overline{\mathrm{RD}}$よりもLになるのが遅い。
※2：この長さは（M1サイクル時の$\overline{\mathrm{MREQ}}$）<（通常の$\overline{\mathrm{MREQ}}$）<$\overline{\mathrm{IORQ}}$である。

Instruction Opcode Fetch.

CPUの一連の動作は，まず命令の読み出しに始まるという意味で$\overline{\mathrm{M1}}$サイクル（エムワンサイクル，第一マシンサイクル）とも呼ばれます。

Z-80は$\overline{\mathrm{M1}}$サイクルに入ると，次の順序で命令を読み出す準備をします。

①読み出す命令のある番地（プログラム・カウンタの値）をアドレスバスにセットします。

②アドレスバス確定後，$\overline{\mathrm{M1}}$をLにして，CPUが$\overline{\mathrm{M1}}$サイクルに入ったことを外部に知らせます。

③続いて，$\overline{\mathrm{MREQ}}$と$\overline{\mathrm{RD}}$をほぼ同時（メーカーも、どちらかが先になるかは言明していません）にLにします。

◇

以上で，メモリから読みだす準備はできましたので，この後ϕに与えられているクロックのほぼ1周期（ϕが2MHzなら約500ns）分待ったあと，データバスを読みます。

続いてCPUは命令のデコード（解釈）に入るわけですが，この間にZ-80は，$\overline{\mathrm{RFSH}}$＝Lとして，ダイナミックRAM用のリフレッシュに入るわけです。リフレッシュは，$\overline{\mathrm{M1}}$サイクルの直後に必ず行なわれます。

メモリが低速で，CPUが読み込むまでにプログラムが出てこないときは，$\overline{\mathrm{WAIT}}$＝Lとすることで，CPUに待ってもらうことができます。CPUは，$\overline{\mathrm{RD}}$＝$\overline{\mathrm{MREQ}}$＝Lとなった後のクロックの立ち下がりで，$\overline{\mathrm{WAIT}}$端子をセンスしており，もし$\overline{\mathrm{WAIT}}$＝Lなら，次のクロックの立ち下がりまで待ってふたたび$\overline{\mathrm{WAIT}}$のセンスを行なうということを繰り返します。$\overline{\mathrm{WAIT}}$＝Hなら，データバスを読むわけです。

$\overline{\mathrm{WAIT}}$の扱われ方は，他のサイクルでも同様です。

Memory Read Cycle

$\overline{\mathrm{M1}}$サイクルと同じく，メモリ空間をアクセスするサイクルですが，タイミング的には$\overline{\mathrm{M1}}$よりも余裕を持たせたものとなっていますが，大ざっぱなところは，$\overline{\mathrm{M1}}$がLにならない他は似たようなものです。

Memory Write Cycle

メモリ空間への書き込みは次の手順によります。

①アドレスバスの確定。

②データバスに，書き込みたいデータをセットします。

③$\overline{\mathrm{MREQ}}$＝Lとします。

④$\overline{\mathrm{MREQ}}$より1クロックほど遅れて$\overline{\mathrm{WR}}$＝Lとします。

⑤さらに約1クロック後に$\overline{\mathrm{WR}}$＝$\overline{\mathrm{MREQ}}$＝Hとして書き込み動作を終了します。

Input/Output Cycle

基本的には，メモリのリード・ライトと同様ですが，$\overline{MREQ}$の代わりにIORQが使われることと，自動的にウェイトが1回（ワン・ウェイトと言います）挿入される点が異なります。

Interrupt Request /Acknowlege Cycle

CPUは，各リード・ライトサイクルの最後の，クロックの立ち上がりで$\overline{INT}$端子をセンスしています。

もし，$\overline{INT}$＝Lとなっているとき CPUは特別なM1サイクルに入ります。

CPUは，$\overline{M1}$をLにしてから（2クロック半ほど遅れて）$\overline{IORQ}$＝Lとしてから，データバスを読みます。

このサイクルが特殊なのは，$\overline{RD}$も$\overline{WR}$も変化しないことです。

外部では，$\overline{M1}$＝$\overline{IORQ}$＝Lを，割り込み要求に対する応答として扱います。

◇

ここで読み出したデータが，CPU内でどのように扱われるかについて少し書いておきます。直接，ハードウェアとは関係ありませんので無視して，次に進まれてもけっこうです。

Z-80には3つの割り込み時の動作モードがあり，ソフトウェアによってセットされます（IM0，IM1，IM2の各命令）。

①モード0

このモードは，Z-80の前進であるインテル社の i8080 と同じで，読み込んだデータは命令として扱われます。通常は，1バイトのジャンプ命令（RST命令）を読み込ませるような周辺回路を組んで使います。

②モード1

このモードでは，読み込んだデータは使用されず，0038番地にジャンプします。

③モード2

Z-80でもっとも強力な割り込みで，Iレジスタを上位8bit，読み込んだデータを下位8bitとするアドレスにジャンプします。

◇

なお，CPUがリセットされると，モード0に自動的にセットされます。

図2　Instruction Opcode Fetch

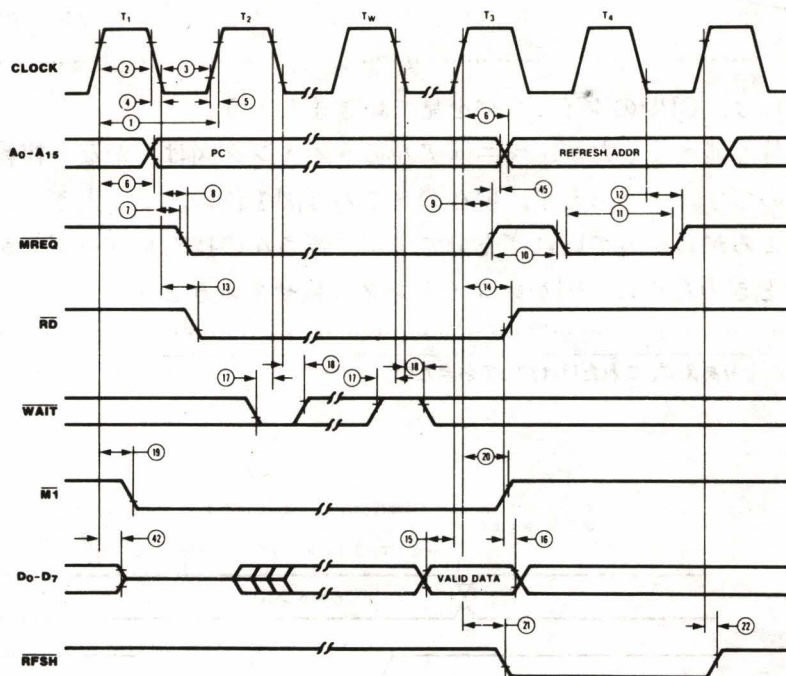
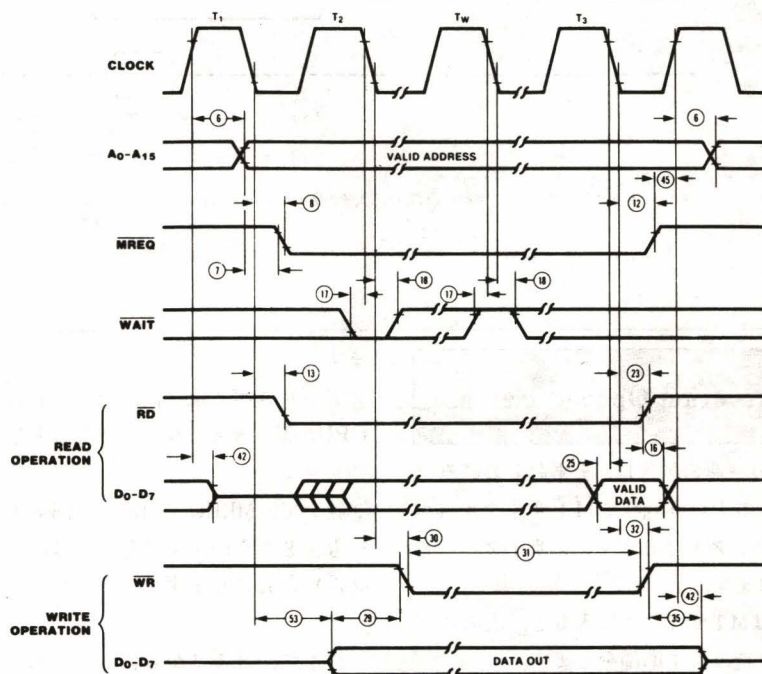



図3　Memory Read or Write Cycle

Non-Maskable Interrupt Request Cycle

$\overline{INT}$と同様に，$\overline{NMI}$も各リード・ライトサイクルの最後のクロックの立ち上がり時にセンスされます。

$\overline{NMI}$＝Lが検出されると，0066番地にジャンプします。

Bus Request /Acknowlege Cycle

$\overline{BUSRQ}$も$\overline{INT}$や$\overline{NMI}$と同様なタイミングでセンスされます。

$\overline{BUSRQ}$＝Lを検出すると，アドレスバスやデータバス，$\overline{MREQ}$，$\overline{IORQ}$，RD，WRをすべて，ハイ・インピーダンス状態としたあと，$\overline{BUSAK}$＝Lとして，$\overline{BUSRQ}$＝Hとなるのを待ちます。$\overline{BUSRQ}$の解除はクロックの立ち上がりでセンスされます。

Halt Acknowlege Cycle

CPUがHalt命令を読み込むと，$\overline{HALT}$＝Lとして停止します。この状態を解除できるのは，$\overline{INT}$，$\overline{NMI}$，$\overline{RESET}$です。

Reset Cycle

$\overline{INT}$，$\overline{NMI}$，$\overline{BUSRQ}$と同じタイミングでセンスされ，$\overline{RESET}$＝Lだと，アドレスバス，データバスをハイ・インピーダンス，その他をHとして，$\overline{RESET}$＝Hとなるのを待ちます。

$\overline{RESET}$＝Hで，0000番地よりプログラムを実行していきます。

図4 Input/Output Cycle

図5 Interrupt Request/Acknowlege Cycle

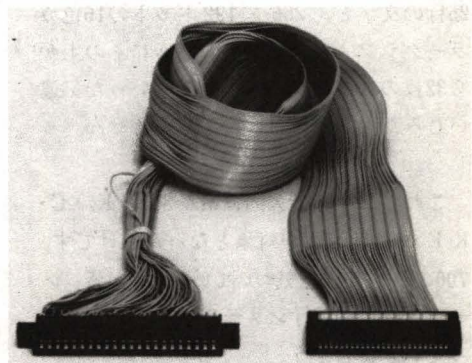
MZ-700用コネクタの製作

ハードウェア入門は，MZ-80K/C/1200用としてやってきましたが，『MZ-700用にならないか』という声が多いようですので，ここらで検討してみることにします。

MZ-700シリーズは，もともとK/C/1200の改良版であり，シャープもK/Cの周辺機器が使えると言っているようですから，一応ハード的なコンパチビリティはある程度保たれていると思い，調べてみました。

K/Cシリーズと700の違いのうち，この連載と関係のありそうな点は次の4つでした。

①$ 0000～$ 0FFF，および$ D000～$ FFFFに裏RAMが存在する。

②I/Oマップの$ FEおよび$ FFがプ

リンタに使用されている。

③クロック周波数が異なる（K/Cは2MHz，700は約3.6MHz)。

④拡張バスがK/Cではコネクタであったのが，700ではカードエッジに変更され，引き出されている信号線が増えた。

このうち①から③は，私のほうで気をつければいいことですから，皆さんに関係するのは④のみです。

それでは，700用の接続ケーブルを作ることにしましょう。

大まかなところは第1回（3月号）と同じですから，そちらを見てもらうことにして，ここでは変更点のみ書いておきます。

○フラットケーブル・コネクタにヒロセのH1F5A-50D2.54Rなどのカードエッジ用のものを使います。

○続いて，3月号の作業行程表の前に，次の作業を追加してください。

○F22は，どこにもつながないようにする(短く切っておくとよいでしょう)。

○F17-A2

○F24-B19

○F30-B2

○F50-B3

以下は，3月号に従ってください。くれぐれもF22をGNDにつないだりしないように。

コネクタができましたので，さっそく6月号のLEDボードを接続してみましたが特に問題はないようで，プログラムもそのままでOKでした。

河合さんが3月号に発表したグラフィックボードもそのまま使うことができますから，うまくすると9918のスーパーインポーズ機能によって，PC-8801のように40字×25行の文字と，256×192ドットの16色カラーグラフィックを重ね，さらにその上から32枚のスプライトを重ねるといった気違いじみた画面が作れるかもしれません。

これで，ハード入門はMZ-80K・K2・C・K2E・1200・700が対象となったわけです。700のみが若干色気づいていますが，皆，シャープの40字25行シリーズです。仲良くやっていきましょう。

図6 Non-Maskable Interrupt Request Cycle

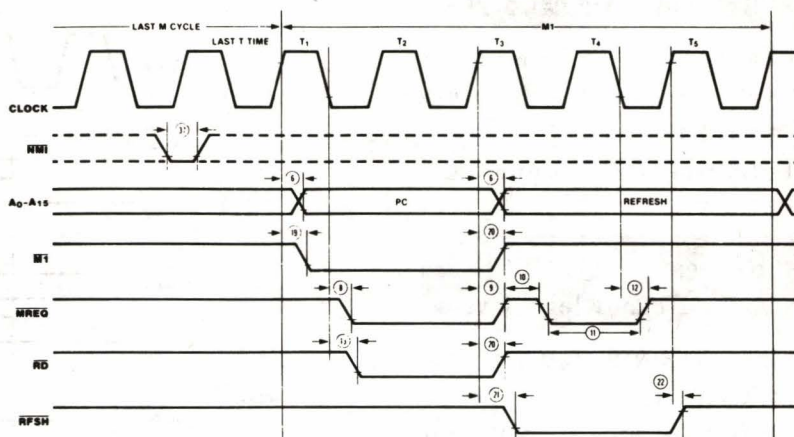



図7 Bus Request/Acknowledge Cycle

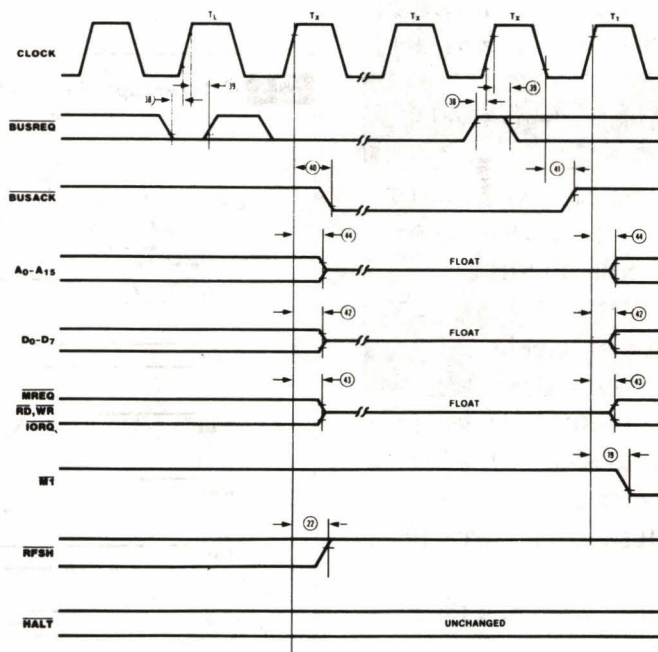



図8 Halt Acknowledge Cycle

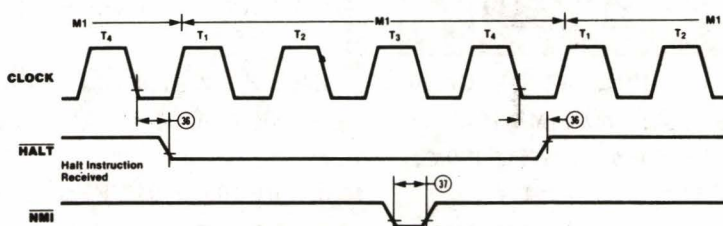
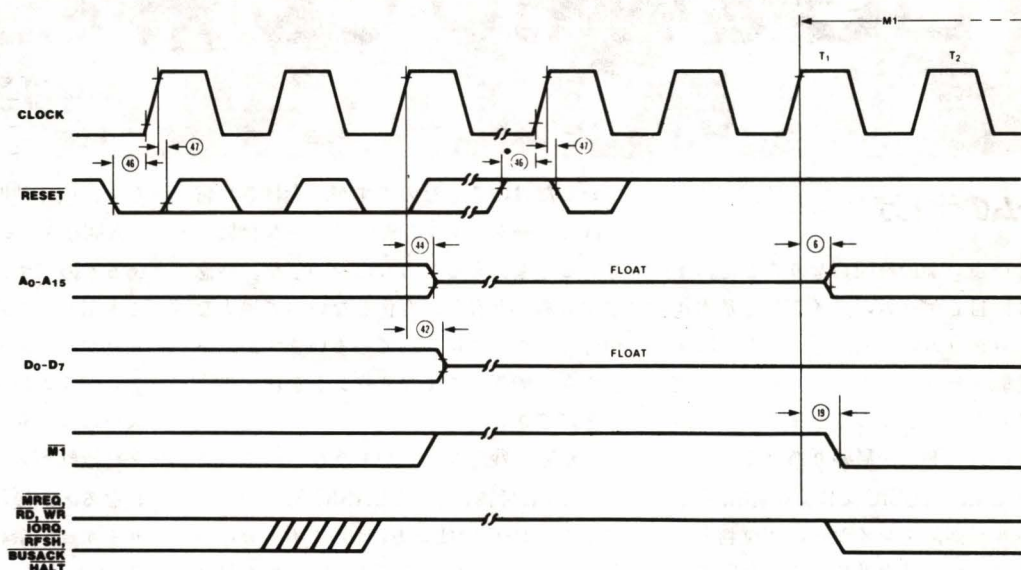
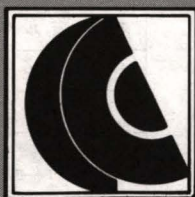



図9 Reset Cycle

X1(HuBASIC) G RAM不要 アイデアから仕上げまで

PSGを利用して効果を上げる

Elder Man

●ゲームの仕上げ

本稿もいよいよ最終回になりました。1回目，2回目でゲームのアイデアと基本ルーチンプログラムの作成，さらにPCG（ユーザー定義キャラクタジェネレータ）を使用してプログラムを作ってきました。今回はゲームのムードを一層盛りあげるためのSOUND効果，MUSIC文による演出，そしてタイトル画面，プレイヤーの順位判定プログラムについて話を進めます。今回でスーパードライブゲームが仕上がりますので，最後に完成品としてのプログラムリストを載せました。

今月号からの読者のために，ゲームのアイデアについてもう一度説明します。

BASICプログラムでリアルタイムのドライブゲームを作ることをテーマにしています。ドライブゲームというのは，プレイヤーの操作する自動車がＴＶ画面の下にあって，上から下へとくねくねと流れる道路上を走行させてタイムを競うものです。プレイヤーはアクセルとハンドルを操って，道路上を運転し，他の自動車にぶつからないよううまく走り，タイムトライアルに挑戦します。タイムトライアルをおもしろくするために，ベストタイムを記録したプレイヤーの中間タイムを刻一刻と表示して，新記録達成のスリルを味わおうというアイデアを入れています。

●X1のPSG

Ｘ1の特徴としてPCG（プログラマブルキャラクタジェネレータ）とPSG（プログラマブルサウンドジェネレータ）が標準装備されている点があります。PCGのプログラミングについては前回でやりましたので，今回はPSGの使い方について話を進めます。PSGというのは小さな1個のLSIが付属しているだけのことなのですが，大げさに言えばミュージックシンセサイザーが付属しているとも言えるしろものなのです。残念ながらあまり音色が変化しないので過大な期待をもつと裏切られてしまいますが，それでも結構さまざまなサウンドを作り出すことができます。

しかし，残念なことがもうひとつあります。Ｘ1に付属しているBASIC MANUALにはこのPSGの使い方について不十分な説明しかされていないのです。実は私も最初は何のことだか，どう使えばよいのかまったくわからなくて使用する気がしなかったものです。しかし，天の助けというか，私の友人に一時期シンセサイザーのオモチャ（失礼！）をいじっているのがいまして，彼によってＸ1のPSGの使用方法が解明されたのです。持つべきものは友か！。全国のマイコン族よ，ひとりでコツコツやるだけでなく，広く仲間と交流しようではないか！

人間ひとりの能力というものには限りがあるものです。仲間があれば，自分ではとても気がつきそうにもなかったことなど新発見が多くあると思います。他人の作ったプログラムのリストをキー入力することも大いに役に立ちます。他人のプログラミングに無駄が多いなんてことに気がつくようになると，優越感をもったりすることがありますョ。余談はともかくとして，SOUND文の使い方について話を戻します。まず表1をみてください。 付属のBASIC MANUALに載っている表です。ビットとして7から0までの数字がかかれていますが，これは8ビットのことで右から左へと数字が大きくなります。10進数でいうと0から255までが表現できます。これがデータです。レジスタ番号が0から13まであります。レジスタというのが具体的に何かという点につ

表1　ＰＳＧのレジスタとデータの関係

レジスタ番号	レジスタの内容 ＼ ビット	7	6	5	4	3	2	1	0
0	チャンネルＡの周波数	8ビット FTA							
1	(音程)	未使用				4ビット CTA			
2	チャンネルＢの周波数	8ビット FTB							
3	(音程)	未使用				4ビット CTB			
4	チャンネルＣの周波数	8ビット FTC							
5	(音程)	未使用				4ビット CTC			
6	ノイズの周波数	未使用			5ビット データ				
7	I/Oポート，ミキサ切換	IN/OUT IOB	IN/OUT IOA	ノイズ C	ノイズ B	ノイズ A	トーン C	トーン B	トーン A
8	チャンネルＡの音量	未使用			M	L3	L2	L1	L0
9	チャンネルＢの音量	未使用			M	L3	L2	L1	L0
10	チャンネルＣの音量	未使用			M	L3	L2	L1	L0
11	エンベロープ周期	8ビット FT							
12		8ビット CT							
13	エンベロープ形状	未使用				E3	E2	E1	E0

いては私にはわかりませんが，とにかく0～13番の窓口があると考えていて十分だと思います。レジスタと呼ばれる窓口にデータを送り込むことによってサウンドを作ったり，変化させたりするわけです。各レジスタについて，その使い方を説明します。

レジスタNO. 0～1　2つのレジスタでチャンネルAの周波数(音程)を設定するようになっています。チャンネルはA，B，C 3チャンネルあり，まったく別々に3つのチャンネルから音を同時に出せるということです。レジスタNO.0は，8ビットの全部を使い，レジスタNO.1は下4ビットを使うことになっています。NO.0のレジスタに入れる値をFTAとし，NO.1のレジスタに入れる値をCTAとすると，チャンネルAから出てくる音の周波数(Hz)は次の式によって求められます。

$$F(A) = 125,000/(CTA*256+FTA)$$

念のため，FTAは0～255，CTAは0～15の範囲の値になります。

レジスタNO.2～5　チャンネルB，C用で，チャンネルAと同じです。

レジスタNO. 6　ノイズというのは「ザーッ」という音で，このレジスタではそのノイズの音の高さを指定します。下5ビットですから0～31の範囲で指定します。0が最高音で，31が最低音になります。

レジスタNO.7　I/Oポート，ミキサの切り換えになっています。6，7ビットは関係がないそうです。つまり，下6ビットでA～Cチャンネルから出す音を澄んだトーンにするか，ノイズにするか，一緒に出すかを指定します。ビットごとにみると0か1かということになりますが，0で出力，1でOFFになっています。63というデータを入れれば3つのチャンネルがOFFになります(2進数で00111111)。

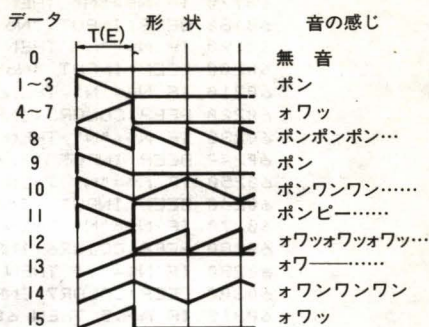
レジスタNO. 8～10　A～Cチャンネルの音量の設定とエンベロープの使用，未使用の設定を行ないます。下4ビットで音量を設定します。15で最大音量，0で最小音量(つまりOFF)になります。エンベロープを使う場合は4ビットのところを1とし，使用しない場合は0にします。エンベロープというのは波形のことで，これにより音色を変化させることができます。波形の形についての設定はレジスタNO.13で行ないます。エンベロープを使わないとフルートの音のような正弦波が持続的に出るだけですから，効果的などの音作りには不向きです。音楽演奏のためのMUSIC文，PLAY文での音はこの正弦波の音を使用しています。MUSIC文もPLAY文もこのPSGを使用していて，文字変数などの入力で音階をとれるようになっている命令ですから，別なところから音を出しているわけではありません。つまりSOUND文を使ったあとはノイズの使用などキャンセルしておかないと，あとで音楽を演奏しようとしてもまともな音色にならないことがあるので要注意！

レジスタNO.11～12　エンベロープの周期を設定します。レジスタNO.11の値をFTとし，レジスタNO.12の値をCTとするとエンベロープの周期(秒)は次の式によって求められます。

$$T(E) = (CT*256+FT)/7800$$

FT＝185，CT＝30で約1秒の周期になります。また，エンベロープは，A～Cチャンネルの全部を支配するので，チャンネルごとの独立した使用，不使用の区別はできません。

レジスタNO.13　下位4ビット0～15の範囲でエンベロープの形状の指定をします。エンベロープの形状は次のようになっています。

図1　エンベロープ形状

●PSG CREATER

長々とPSGのレジスタとビットデータの説明をしましたが，ひとつの効果音を作るのにいちいち対照表とにらめっこでは頭が混乱するし，作りたくなくなってしまいます。そこで，このようなめんどうなことはコンピュータにまかせて，人間様は気楽に音作りができないかということを考えつくことになります。PCGについても，PSGについてもそれらを有効に活用するためには開発用ツールと呼ばれるソフトの利用が一番賢明な方法だと思います（私の場合は，ゲームプログラムにとりかかる前にこのような開発用ツールプログラムを作っています)。しかし，現在まだSOUND文の開発ツールソフトがあるということは知りませんので，私が効果音を作るときに利用している，簡単な音色を調べるためのプログラムを紹介します。プログラム1として紹介したPSG CREATERは，これまでの説明を一応理解していればマニュアルや説明文を読まなくても音作りができると思います。

本当にPSG CREATERと呼べるようなソフトはもっと複雑な波形まで扱えなければいけないと思っています。どなたか，よい開発用ツールプログラムの作成に挑戦してみませんか。

リスト1

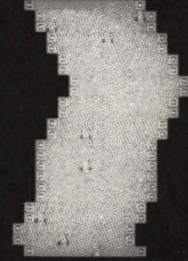
```
60000 REM ****************************************************
60010 REM
60020 REM  PSG CREATER   Copyright 1983 by Elder Man  Vr.1.0
60030 REM
60040 REM ****************************************************
60050 WIDTH 80:COLOR 5,0:CLICKOFF
60060 LOCATE 0,0:PRINT "***  P.S.G. CREATER  by  Elder Man  1983  Vr.1.0  ***"
60070 COLOR 7:PRINT:PRINT"SOUND DATA INPUT":CLEAR
60080 BEEP:COLOR6:INPUT "A ch FT (<4096) = ",A:A(1)=INT(A/256)
60090 IF A>255 THEN A(2)=(A MOD 256) ELSE A(2)=A
60100 BEEP:INPUT "B ch FT (<4096) = ",B:B(1)=INT(B/256)
60110 IF B>255 THEN B(2)=(B MOD 256) ELSE B(2)=B
60120 BEEP:INPUT "C ch FT (<4096) = ",C:C(1)=INT(C/256)
60130 IF C>255 THEN C(2)=(C MOD 256) ELSE C(2)=C
60140 BEEP:COLOR 7:INPUT "Noise FT (<31) = ",M
60150 IF M>31 THEN 60140
```

```
60160 BEEP:INPUT "Noise A ch ? (Y/N) = ",N$
60170 IF N$="N" THEN N=8
60180 BEEP:INPUT "Noise B ch ? (Y/N) = ",N$
60190 IF N$="N" THEN N=N+16
60200 BEEP:INPUT "Noise C ch ? (Y/N) = ",N$
60210 IF N$="N" THEN N=N+32
60220 BEEP:COLOR5:INPUT "Tone A ch ? (Y/N) = ",N$
60230 IF N$="N" THEN N=N+1
60240 BEEP:INPUT "Tone B ch ? (Y/N) = ",N$
60250 IF N$="N" THEN N=N+2
60260 BEEP:INPUT "Tone C ch ? (Y/N) = ",N$
60270 IF N$="N" THEN N=N+4
60280 BEEP:COLOR6:INPUT "Envelope A ch ? (Y/N) = ",N$
60290 IF N$="Y" THEN A(3)=16
60300 BEEP:COLOR7:INPUT "Volume A ch (<15) = ",A
60310 IF A>15 THEN 60300 ELSE A(3)=A(3)+A
60320 BEEP:COLOR6:INPUT "Envelope B ch ? (Y/N) = ",N$
60330 IF N$="Y" THEN B(3)=16
60340 BEEP:COLOR7:INPUT "Volume B ch (<15) = ",B
60350 IF B>15 THEN 60340 ELSE B(3)=B(3)+B
60360 BEEP:COLOR6:INPUT "Envelope C ch ? (Y/N) = ",N$
60370 IF N$="Y" THEN C(3)=16
60380 BEEP:COLOR7:INPUT "Volume C ch (<15) = ",C
60390 IF C>15 THEN 60380 ELSE C(3)=C(3)+C
60400 BEEP:COLOR6:INPUT "Envelope FT-lo (<255) = ",E(1)
60410 IF FT>255 THEN 60400
60420 BEEP:INPUT "Envelope CT-hi (<255) = ",E(2)
60430 IF FT>255 THEN 60420
60440 BEEP:INPUT "Envelope Form (<15) = ",E(3)
60450 IF FT>255 THEN 60440
60460 SOUND 0,A(2):SOUND 1,A(1):SOUND 2,B(2):SOUND 3,B(1)
60470 SOUND 4,C(2):SOUND 5,C(1):SOUND 6,M:SOUND 7,N
60480 SOUND 8,A(3):SOUND 9,B(3):SOUND 10,C(3)
60490 SOUND 11,E(1):SOUND 12,E(2):SOUND 13,E(3)
60500 LOCATE0,22:PRINTCHR$(5):LOCATE0,23:PRINTCHR$(5)
60510 COLOR7:LOCATE 0,22:PRINT A(2);A(1);B(2);B(1);C(2);C(1);M;N
60520 PRINT A(3);B(3);C(3);E(1);E(2);E(3)
60530 S$=INKEY$:IF S$="" THEN 60530
60540 IF S$="S" THEN SOUND 8,0:SOUND 9,0:SOUND 10,0:COLOR 5:GOTO 60060
60550 IF S$="R" THEN 60460
60560 GOTO 60530
```

リスト2

```
10 '***********************************************************************
20 '
30 '  SUPER DRIVE   for  X1    by - Elder Man     1983.3.15     Vr.3.0
40 '
50 '***********************************************************************
51 '==== ｼｮｷ ｾｯﾃｲ ====
53 WIDTH 40:INIT:CGEN:COLOR 7,0:DEFINT A-Y:TEMPO 260:CLICK OFF
54 DIM B(200),C(200)
55 GOSUB 9500
56 GOSUB 9000:RESTORE 340
60 PX=15:U=1:N=200:V=0:Z=0:COLOR 5
70 LOCATE 0,0:PRINT "* DRIVE * LAP=     TIME=     SPEED=     "
80 COLOR7:CGEN1:FOR I=1 TO 24:LOCATE 10,I:PRINT "61111111116";
85 SOUND 0,160:SOUND 1,15:SOUND 2,208:SOUND 3,7:SOUND 4,232:SOUND 5,3
86 SOUND 6,31:SOUND 7,56:SOUND 8,31:SOUND 9,31:SOUND 10,31
87 SOUND 11,255:SOUND 12,0:SOUND 13,8
90 NEXT:BEEP:TIME=0
99 '==== ﾒｲﾝ ﾙｰﾁﾝ ====
100 CGEN 0:LOCATE 14,0:PRINT B(V):LOCATE24,0:PRINT TIME
110 LOCATE 35,0:PRINT 230-N:C(V)=TIME:SOUND 11,N+55
120 LOCATE0,1:PRINT CHR$(&HF)
130 READ A:IF A=99 THEN 1000
140 CGEN 1:LOCATEA,1:PRINT"61111111116"
150 IF RND<.8 THEN 180
160 IF RND>.6 THEN LOCATE A+1+RND*8,1:PRINT "5":GOTO 180
170 LOCATE A+1+RND*7,1:PRINT "7"
180 IF CHARACTER$(PX,24)<>"1" GOSUB 250
190 S=STICK(0):ON S GOSUB 290,300,310,320,330,335
200 LOCATE PX,24:PRINT"2";
210 IF N<0 THEN N=0
220 FOR I=0 TO N*4:NEXT
230 N=N+2:IF N>200 THEN N=200
240 Z=Z+.2:V=Z:GOTO 100
```

```
249 '==== ハ゛クハツ ヒョウシ゛ ====
250 FOR J=0 TO 1
260 LOCATE PX,24:PRINT "3";:SOUND 6,31:SOUND 7,7:SOUND 8,31:SOUND 9,31
265 SOUND 10,31:SOUND 11,185:SOUND 12,20:SOUND 13,9:PAUSE 1
270 LOCATE PX,24:PRINT "4";:SOUND 6,31:SOUND 7,7:SOUND 8,31:SOUND 9,31
275 SOUND 10,31:SOUND 11,185:SOUND 12,20:SOUND 13,9:PAUSE 1
280 NEXT:N=200
282 SOUND 0,160:SOUND 1,15:SOUND 2,208:SOUND 3,7:SOUND 4,232:SOUND 5,3
284 SOUND 6,31:SOUND 7,56:SOUND 8,31:SOUND 9,31:SOUND 10,31
285 SOUND 11,255:SOUND 12,0:SOUND 13,8
286 RETURN
289 '==== クルマ ノ ソウサ ====
290 PX=PX-1:IF PX<0 THEN PX=0
295 RETURN
300 N=N+20:RETURN
310 PX=PX+1:IF PX>37 THEN PX=37
320 RETURN
330 N=N-10:RETURN
335 CGEN0:SOUND 8,0:SOUND 9,0:SOUND 10,0:BEEP:GOTO 1010
339 '==== ト゛ウロ テ゛ータ ====
340 DATA 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10
350 DATA11,11,11,11,11,12,12,12,12,12,12,13,13,13,14,14,14,14,15,15,15,16
360 DATA 16,16,17,17,18,19,20,21,22,23,24,25,26,27,28,28,28,28,27,27,27,27
370 DATA 28,28,28,28,28,28,28,28,27,27,27,26,26,26,26,25,25,24,23,22,21,20
375 DATA 19,18,17,16,15,14,13,12,11,10,9,8,7,6,5,4,3,2,1,0,0,1,2,3,3,4,4,5
380 DATA 4,3,2,1,0,0,0,0,0,0,0,0,0,1,1,2,3,4,5,5,6,6,7,7,8,8,9,9,10,10,10,10
390 DATA 10,10,10,10,10,10,11,11,12,12,13,13,12,12,11,11,10,10,9,9,8,8,9,9
400 DATA 10,10,10,10,10,10,11,11,12,12,13,13,12,12,11,11,10,10,9,9,8,8,9,9
410 DATA 10,11,12,13,14,15,16,17,18,19,20,22,24,26,28,27,27,26,25,25,24,24
420 DATA 23,24,23,22,21,21,20,20,18,16,14,13,12,12,11,11,10,10,9,9,8,8,9,9
430 DATA 10,10,10,10,10,10,11,11,12,12,13,13,12,12,11,11,10,10,9,9,8,8,9,9
440 DATA 10,10,10,10,10,10,11,11,12,12,13,13,12,12,11,11,10,10,9,9,8,8,9,9
450 DATA 10,11,12,12,11,10,9,8,8,7,7,6,6,5,5,4,3,2,1,0,2,4,6,7,8,9,8,8,9,9
460 DATA 10,10,10,10,10,10,11,11,12,12,13,13,12,12,11,11,10,10,9,9,8,8,9,9
470 DATA 10,10,10,10,10,10,11,11,12,12,13,13,12,12,11,11,10,10,9,9,8,8,9,9
480 DATA 11,13,15,15,15,15,16,16,16,15,15,15,16,16,18,20,22,24,26,28,28,28
490 DATA 27,27,26,26,26,25,25,24,24,23,22,21,20,20,19,18,17,17,16,16,16,15
500 DATA 14,13,13,12,12,11,11,11,12,12,13,13,12,12,11,11,10,10,9,9,8,8,9,9
510 DATA 8,8,7,6,6,5,4,3,2,0,0,0,0,2,4,6,8,10,12,11,11,11,10,9,9,9,8,8,9,9
520 DATA 10,10,10,10,10,10,11,11,12,12,13,13,12,12,11,11,10,10,9,9,8,8,9,9
530 DATA 10,10,10,10,10,10,11,11,12,12,13,13,12,12,11,11,10,10,9,9,8,8,9,9
540 DATA 10,10,10,10,10,10,11,11,12,12,13,13,12,12,11,11,10,10,9,9,8,8,9,9
550 DATA 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10
560 DATA 11,11,11,11,11,12,12,12,12,12,12,13,13,13,14,14,14,14,15,15,15,16
570 DATA 16,16,17,17,18,19,20,21,22,23,24,25,26,27,28,28,28,28,27,27,27,27
580 DATA 28,28,28,28,28,28,28,28,27,27,27,26,26,26,26,25,25,24,23,22,21,20
585 DATA 19,17,15,13,11,9,7,6,5,4,4,3,3,2,2,2,1,1,1,1,2,2,2,3,3,3,4,4,5,5,5
590 DATA 4,3,2,1,0,0,0,0,0,0,0,0,0,1,1,2,3,4,5,5,6,6,7,7,8,8,9,9,10,10,10,10
600 DATA 10,10,10,10,10,10,11,11,12,12,13,13,12,12,11,11,10,10,9,9,8,8,9,9
610 DATA 10,10,10,10,10,10,11,11,12,12,13,13,12,12,11,11,10,10,9,9,8,8,9,9
620 DATA 10,11,12,13,12,11,10,11,12,14,16,18,20,22,24,26,28,28,28,26,26,25
630 DATA 24,23,22,22,22,21,20,21,22,23,23,22,21,20,18,16,14,12,10,8,7,6,5,6
640 DATA 7,8,9,10,10,10,10,11,11,12,12,13,13,12,12,11,11,10,10,9,9,8,8,9,9
650 DATA 10,10,10,10,10,10,11,11,12,12,13,13,12,12,11,11,10,10,9,9,8,8,9,9
660 DATA 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10
670 DATA 11,12,14,16,18,20,22,24,22,20,18,16,14,12,11,10,10,10,10,10,10,10
680 DATA 9,9,9,8,8,8,9,9,9,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10
690 DATA 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10
700 DATA 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10
710 DATA 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10
720 DATA 10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,99
999 '==== TRY AGAIN ? ====
1000 IF C(V)<B(V) GOSUB 1200
1002 SOUND 8,0:SOUND 9,0:SOUND 10,0
1005 GOSUB 2000
1010 S$=INKEY$:IF S$="" THEN 1020 ELSE 1010
1020 CGEN0:LOCATE 8,10:PRINT "TRY AGAIN ? (Y/N)
1030 S$=INKEY$:IF S$="Y" OR S$="y" THEN 56
1040 IF S$="N" OR S$="n" THEN END ELSE 1030
1200 FOR I=0 TO V:B(I)=C(I):NEXT:RETURN
1999 '==== ナンイ カナ ? ====
2000 FOR J=0 TO 5
2010 IF C(V)<D(J) THEN 3000
2020 NEXT:GOSUB 4600:RETURN
2999 '==== アナタ ノ ナマエ ハ ? ====
3000 ON J+1 GOSUB 4000,4100,4200,4300,4400,4500
3002 FOR I=5 TO J STEP -1:D$(I+1)=D$(I):D(I+1)=D(I):NEXT:D(J)=C(V)
3005 S$=INKEY$:IF S$<>"" THEN 3005
3010 LOCATE 0,12:PRINT SPC(38):BEEP
3020 LOCATE 0,12:INPUT "YOUR NAME = ";D$(J)
3030 RETURN
4000 MUSIC "V14O4C2EGECEGECEGECEG+C8":RETURN
```

```
4100 MUSIC "V1404C2DEFGFEDCDEFGFEDC8":RETURN
4200 MUSIC "V1404C2DEDCDEDCDEDC8":RETURN
4300 MUSIC "V1404C2ECECECECECEC8":RETURN
4400 MUSIC "V1404C2DCDCDCDCDCDC8":RETURN
4500 MUSIC "V1404C2RCRCRCRCRCRC8":RETURN
4600 MUSIC "V1403+C2BAGFEDC8":RETURN
7999 '==== HOW TO PLAY ===
8000 CLS:COLOR 5
8010 LOCATE 0,0:PRINT "****** HOW TO PLAY ******":BEEP
8020 COLOR 7:PRINT:PRINT "  ｷｰ ﾉ ｿｳｻ ﾎｳﾎｳ"
8030 COLOR 6:PRINT:PRINT
8040 PRINT "     ┌─┐┌─┐  5 = ACCELL ( ｶｿｸ )"
8050 PRINT "     |5||6|  6 = RETIRE ( ﾘﾀｲｱ )"
8060 PRINT "     └─┘└─┘"
8070 PRINT " ┌─┐┌─┐┌─┐  1 = LEFT   ( ﾋﾀﾞﾘ )"
8080 PRINT " |1||2||3|  2 = BREAK  ( ﾌﾞﾚｰｷ)"
8090 PRINT " └─┘└─┘└─┘  3 = RIGHT  ( ﾐｷﾞ  )"
8100 COLOR5:PRINT:PRINT:PRINT "LAP   = CHAMP'S TIME ┌ < 150 sec A class";
8110 PRINT:PRINT "TIME  = YOUR TIME    | < 160 sec B class";
8120 PRINT:PRINT "SPEED = CAR SPEED    └ < 180 sec C class";
8130 COLOR1:LOCATE 0,21:PRINT "STAND BY OK ? Hit any Key !":BEEP
8140 S$=INKEY$:IF S$="" THEN 8140
8150 RETURN
8999 '==== ｹﾞｰﾑ ﾉ ｱﾝﾅｲ ====
9000 CLS:COLOR 5,0:CSIZE 3
9010 LOCATE 0,0:PRINT#0 "*** SUPER DRIVE ***"
9020 CSIZE:COLOR3:LOCATE 20,3:PRINT "by Elder Man"
9030 COLOR 7:LOCATE 0,5:PRINT "SPEED DRIVE RECORDS"
9040 COLOR 5:PRINT:PRINT "CHAMP"
9050 PRINT:PRINT "2 nd"
9060 PRINT:PRINT "3 rd"
9070 PRINT:PRINT "4 th"
9080 PRINT:PRINT "5 th"
9090 PRINT:PRINT "6 th"
9100 FOR I=0 TO 5
9110 COLORI+2:LOCATE 6,I*2+7:PRINT D$(I)
9120 LOCATE 18,I*2+7:PRINT D(I);"sec"
9125 LOCATE 28,I*2+7:PRINT USING "####.#";29255!/D(I);
9126 PRINT " Km/H"
9130 NEXT:COLOR 1:GOSUB 20000
9140 LOCATE 0,20:PRINT "EXPLAIN ? Hit E Key !":BEEP
9150 PRINT:PRINT "Stand by OK ? Hit Other Key "
9480 S$=INKEY$:IF S$="" THEN 9480
9485 IF S$="E" OR S$="e" GOSUB 8000
9490 CLS:RETURN
9499 '==== DEF CHR$ ====
9500 CLS:LOCATE 12,12:PRINT "JUST A MOMENT !":BEEP
9510 RESTORE 10000
9520 FOR I=49 TO 55:A$="":FOR J=1 TO 24
9530 READ A:A$=A$+CHR$(A):NEXT:DEF CHR$(I)=A$:NEXT
9540 FOR I=0 TO 200:B(I)=I*.81+1:NEXT
9550 D$(0)="NIKI LAUDA      ":D(0)=161
9560 D$(1)="M.ANDRETTI      ":D(1)=164
9570 D$(2)="J.STUART        ":D(2)=169
9580 D$(3)="T.IKUZAWA       ":D(3)=174
9590 D$(4)="K.TAKAHASHI     ":D(4)=175
9600 D$(5)="ELDER MAN       ":D(5)=178
9650 CLS:RETURN
9999 '==== PCG ﾃﾞｰﾀ ====
10000  DATA 170,85,170,85,170,85,170,85,0,0,0,0,0,0,0,0,255,255,255,255,255,255,
255,255
10010  DATA 184,61,184,93,186,28,60,28,66,66,66,0,0,219,219,195,165,129,165,231,
231,60,0,36
10020  DATA 170,65,130,64,128,65,128,20,81,150,104,157,106,148,42,201,174,105,15
1,98,149,107,213,54
10030  DATA 168,21,138,85,162,85,42,84,82,74,37,170,92,42,212,43,173,181,218,85,
163,213,43,212
10040  DATA 184,61,184,93,186,28,60,28,90,126,90,24,24,219,231,219,165,129,165,2
31,231,60,0,36
10050 DATA 255,255,195,219,219,195,255,255,0,0,60,60,60,60,0,0,0,126,66,90,90,66
,126,0
10060  DATA 160,1,160,69,162,28,0,4,90,126,90,24,24,219,255,219,189,189,189,255,
255,60,36,60
19999 '==== MUSIC ﾙｰﾁﾝ ====
20000 TEMPO300
20010 MUSIC"V1405G6E3G6E3C6D3E6F3G4R1G4R1G4R1+C4R1G8R5:V1404C5GEGCGEGCGEGC-G4R1G
4R1R5"
20020 MUSIC"V1405A6F3A6F3E6C3E6C3G4R1G4R1-G4R1-G4R1C8:V1404C5AFACGEG-BGD-BC-C4R1
+C4R1"
20030 RETURN
```

●サウンド効果のプログラム

プログラム2のスーパードライブゲームの中には，このPSG CREATERでいろいろ試行錯誤ののちにできあがった音を入れてあります。

85〜87行ではプレイヤーの自動車のアイドリング状態のサウンドを出しています。

110行では，自動車のスピードに応じてエグゾーストノートを変化させています。250〜286行の爆発表示用サブルーチンでは，爆発音を作っています。

335行ではA〜Cチャンネルの出力をOFFにしています。

●ゲームのタイトル画面

プログラム2の9000〜9490行がゲームのタイトル画面で，8000〜8150行がキーの操作の説明のサブルーチンになっています。

タイトル画面をおもしろくするために，F1レーサーの名前とタイム，平均走行速度を表示するようにしたので，それらの変数への代入を9550〜9600行で行なっています。

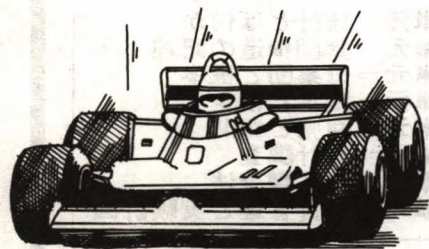
9000行の **CSIZE** というのは，表示する文字の大きさを変える命令で，ゲームのタイトル文字などの表示に便利です。

CSIZE命令はCSIZE0〜3の4通りあり，次のようになっています。

CSIZE 0　ノーマル文字(普通の大きさ)　0は省略できる。
CSIZE 1　垂直2倍文字（縦に2倍)
CSIZE 2　水平2倍文字（横に2倍)
CSIZE 3　垂直2倍，水平2倍文字（縦横とも2倍の大きさ)

CSIZE命令で文字の大きさを決めたあとは，**PRINT**#0命令で画面に表示します。たとえば

CSIZE 3：PRINT#0〝Z″

とすると「Z」という文字が縦横2倍の大きさになって表示されます。ただし，このCSIZE命令を使う場合には使用上の制限があり，それは次のようになっています。

垂直2倍文字………その2行中にノーマル文字や水平2倍文字があってはいけない。

水平2倍文字………必ず偶数のX座標から出力すること。1行中にノーマル文字があっても可。

垂直，水平2倍文字………その2行中にノーマル文字や水平2倍文字があってはいけない。ただし垂直2倍文字はあっても可。必ず偶数のX座標，Y座標から出力すること。

9125行では，自動車の平均走行速度が出せるようにしてあります。29255という値は，ゲームができあがってから最速タイムを求めて逆算しています。

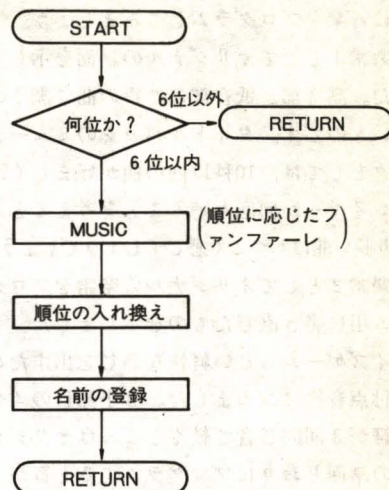
9485行で，キー操作の説明が必要な場合は，HOW TO PLAYのサブルーチンへジャンプするようにしています。〝E″でも〝e″でも受け付けるようにしているのは，CAPS LOCK(大文字入力)していない習慣のユーザーのことも考えてのことです。ゲームの仕上げとなるとこんなことまでもバックアップしておかなければなりません。

ゲームのタイトル画面や説明用の画面はPRINT文でコメントを書くだけですから，プログラミングというよりもカラーとかレイアウトとか作者の趣味性によっています。ただし，ひとつひとつのPRINT文の前にLOCATE命令を使うのはめんどうなこととメモリの節約のためPRINT：PRINTという行あけテクニック（?）を使っています。

●順位づけと名前の登録

1回のプレイが終了するとプログラム2の130行から1000行にジャンプします。2000行から4600行までは，プレイヤーのタイムの順位づけと6位以内に入っていればプレイヤーの名前を登録するサブルーチンになっています。2010行でプレイヤーのタイムC(V)とそれまでの6位までのタイムD(0)，D(1)，D(2)，……，D(5)とを比較しています。6位以内に入らなければ，何もしないでRETURNします。ここのところの処理はすこし複雑なのでフローチャートに描いてみました。

3000行はプレイヤーの順位に対応するファンファーレへの多重分岐文です。

図2　フローチャート

3002行で，順位の入れ換えを行なっています。入れ換えの順は，プレイヤーのタイムが3位になったとすると，いままでの5位を6位に，4位を5位に，3位を4位にして，プレイヤーのタイム，名前を3位に代入します。3002行と3020行のJという変数が順位を表わしていて，これは2010行で使用したJの値を利用しています。

●MUSIC プログラム

ゲームに音楽は欠かせない要素のひとつです。このドライブゲームでは，タイトルバックで流すテーマミュージックとゴールインしたときのファンファーレを入れます。SOUND文に比べると**MUSIC**文も**PLAY**文もたいへんプログラムしやすい命令です。音楽の演奏のテンポは**TEMPO**文で指定します。TEMPO nとして曲の演奏の速さを設定します。nは30〜7500までの値で，1分間あたりの4分音符と拍数を示します。MUSIC文とPLAY文とはほぼ同じ文法，内容の命令ですが，MUSIC文ではテンポの設定をTEMPO文で行なわなければならないところをPLAY文ではPLAY nとすることによりTEMPO nと同じ機能を持ちます。PSGのところで説明しましたが，PSGは3チャンネルあるので，音楽演奏の場合も3つの違った音を同時に発生させることができます。

MUSIC〝チャンネルAのデータ：チャンネルBのデータ：チャンネルCのデータ″

とプログラムすることによって3声（3和音）の音楽演奏ができます。

MUSIC文のBASICの文法上の記法はマニュアルを読んでいただくことにして，実際に音楽をプログラムしてみましょう。

楽譜1としてオリジナルの譜面を示しました。高音部，低音部の2声の曲で調子のよいものです。タイトルバックのミュージックとしては，10秒以内の曲が好ましく，プレイヤーが何回も聴くことを考えるとあまり長い曲はしつこく感じてしまうでしょう。

楽譜2としてオリジナルの楽譜をプログラム用に書き直したものを示しました。ドライブゲームらしい軽快な感じを出すために付点音符に改めました。3小節目の4分音符が3回同じ音で続くところはオリジナルの譜面どおりにプログラムすると音が3つ続けて出てしまいますので，休符を入れることによりオリジナルの曲のイメージを保っています。楽譜2をプログラムしたものが，プログラム2の20000～20020行です。4000～4600行はゴールインしたときの順位に対応させたファンファーレです。

●さいごに

ミュージックプログラムを作っていよいよドライブゲームの完成です。単純なゲームですが，結構難しいことと，タイムトライアルがスリリングなことから楽しめると思います。友人と一緒にタイムトライアルを競うとおもしろさが倍化すると思います。

簡単なプログラムのアイデアから仕上げまで3回に分けて連載させていただきましたが，いかがでしたか。難しいと感じられた方もおられるでしょうし，もっと本格的なプログラムのほうがよかったと言われる方もおられるかも知れません。私もこのような記事は初めてでしたので，あまり上手に進められなかったことをお詫びします。このシリーズについての御意見をいただければありがたいと思います。アンケート葉書で結構ですから，編集部のほうへお寄せください。

●今月のポイント●

○X1に標準装備のPCGとPSG
○マイコン族よ，幅広く交流しよう！
○ゲームの効果音はSOUND文で！
○SOUNDはエンベロープで音色の変化
○面倒なことはコンピュータまかせ
——開発用ソフトの利用——
○タイトル文字はCSIZE文でデッカク表示
○ゲームに欠かせないミュージック

楽譜1　オリジナル

MUSIC by S. OKABE

楽譜2　プログラム用

Arr. by Elder Man

第11回 ―――― 東大MZユーザーズクラブ 有田 隆也

面白玉手箱

MZとX1をもっともっと使いこなそう！

A Mystery Horseを誰か作ってくださいの巻

Horse はHouseの間違いではないかい？ですって。よいのです。暴走しまくる謎の荒馬を意味しているのです。

下のプログラムはMZ-2000とMZ-80B用のプログラムですが，なるべく入力しないことを勧めます。特に，どちらかというと根の暗い方は絶対にスキップしてください。

実はこのプログラム，人を陥れようとして作りかけたものですが，そのようなことをするのにまったく向いていないので，ここまででやめてしまったのです。ですから，これを引きついで，もっと陰険なプログラムを作った方，「玉手箱係」までカセットを送ってください。もっとも陰険なものを紹介したいと思っている次第であります。

万が一，このプログラムを入力する方は，間違いがないかよく確認してからRUNさせてください。何が何だかわからなくなります!?

MYSTERY HORSE for MZ-80B/2000

```
 1 FORI=0TO21:READX:POKE16*16*16*10+I,X:NEXT:Z$="*Error"
 2 Y$="*Error 1 in 40"
 10 DATA 17,0,176,33,162,6,54,255,205,164,6,33,0,176,126,191,254,11,192,195,0,16
0
 50 USR($F14):PRINTY$
 100 USR($A000):GOSUB 20000:IF X$="LIST"THENGOSUB 12000
 900 GOTO100:INPUT X$
 1000 FORI=1TOLEN(X$):PRINTASC(MID$(X$,I,1));:NEXT
 1010 PRINT:GOTO 900
 12000 PRINT" 1 FORI=0TO21:READX:POKE16*16*16*10+I,X:NEXT"
 12002 A=INT(RND(1)*1000):PRINT" 2 ";I=0:J=3
 12003 IFPEEK(A+I)<33 THEN I=I+1:GOTO 12003
 12004 PRINTCHR$(PEEK(A+I));:IF RND(1)<.1 THEN 12007
 12005 I=I+1:GOTO 12003
 12007 PRINT:PRINTJ;" ";:J=J+1:IF RND(1)<.11 THEN PRINT:GOTO 20012
 12008 GOTO 12003
 20000 X$="":I=0
 20010 X$=X$+CHR$(PEEK(16*16*16*11+I)):IF PEEK(16*16*16*11+I+1)=13 THEN RETURN
 20011 I=I+1:GOTO 20010
 20012 PRINT Z$;INT(RND(1)*80+1);"in";1000:RETURN
```

B 地底最大で10万点取る方法の巻

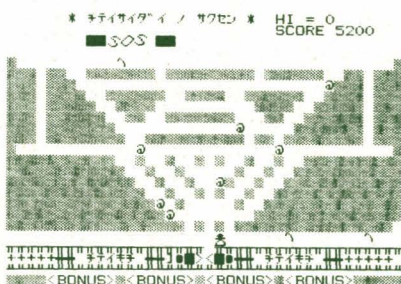
「地底最大の作戦」(本誌2月号掲載)で，なんと，ついに10万点を超えた人が出現したのです。その人は東京にお住まいのOさんで，奥さんとともに日夜血と汗と涙の研究を積んだ結果，ついに超高得点をあげる戦術を発見したのだそうです。

私の目の前で一部を再現してもらいました。そのときの記憶をもとに，私自身がやってみてその画面のハードコピーをとったのが右図であります(ちょっと違うかな？)

シャープのBASICでは，グラフィックとキャラクタの混在コピーがとれないため，やや乱れていますがお許しを。

自分が最初いたところにエサを落としておくこと(図では出ていない)，基地のすぐ上を掘って空間を作っておくこと，階段状の道路を多く作ることなどがコツのようで ⇨

⇨ あります。

いくら世の中広しといえども，Oさんの記録はそうかんたんには破れないと確信しているのですがいかがでしょうか。

C BASICテープのコピー作成についてひとことの巻

X1のBASICテープの予備作成については，マニュアルの195ページに載っていますが，これは付属のテープとまったく同じものを作るものです。ということは，ファンクションキーの割り当ても，初期状態ではマニュアルのようになっています。

ところで，一番手の持っていきやすいF1に「AUTO」が入っているのはいやだ，自分の好きなように初めから割り振っておきたい，という方もいらっしゃるのではないかと思い，やり方を考えてみましたのでご紹介します（ちょっと乱暴なところもありますが……)。

1）BASICテープをLOADする。

2）MON [↵] としてモニタに入る。

3）MFE00 [↵] として次のとおり打ち込む。

```
:FE00=2160FE012000CD41 ↵
:FE08=002A74FEED4B72FE ↵
:FE10=CD4400C303100000 ↵
:FE18=2160FE012000CD3B ↵
:FE20=002A74FEED4B72FE ↵
:FE28=CD3E00C30310     ↵
```

最後に [SHIFT] ＋ [BREAK] でMコマンドから出てください。

4）1)と同じBASIC テープをセットし，GFE00 [↵] とすると，読み込みを開始します（改めて読ませるのは，一度BASICが起動すると，初期化により中身が変わってしまうためです)。

5」D0F42 [↵] とするとファンクションキーの内容が表示されますので，Mコマンドで書き換えます。0F42番地から10HごとにF1からF10までの内容が入っています。まず1バイト目が字数で，その後にアスキーで入れていきます。たとえば，F5を「WIDTH 80 [↵]」にしたいなら，M0F82 [↵] として，

：0F82＝0957494454482038300D [↵]

とします。

6）新しい空テープをセットし，GFE18 [↵] とするとセーブが始まります。

7）カーソルが点滅し始めたら，新しい自分用BASICテープのできあがりです。

㊟2）でモニタに入る前に，WIDTH40になっていることの確認と，4）の前に画面をCLRしておくと画面が乱れません。

D　画面をギッシリ埋めるの巻

MZ-2000, 80B, X1には，画面上に線を引く機能があります。そこで，画面を線でギッシリ埋めてみることにしました。コンピュータゲームの基本にもなり得る迷路を作ってみましょう。

迷路の作り方には，壁を伸ばしていく方法と道を伸ばしていく方法の2つが考えられます。ここでは厚さの無視できる壁を成長させるほうを採用します。迷路内の格子点を順に調べていき，そこに壁がない場合には上下左右の1方向をランダムに選び，その方向に向けて壁を作ります。壁が伸びたところがすでに壁になっている場合には，そこで打ち切りますが，そうでない場合は，また同じようにランダムに壁を伸ばします。この操作をすべての格子点について繰り返すことによって迷路を作成するわけです。

かんたんなのですが，抜け道のひとつしかない迷路を作るために，ちょっとだけ考えなければならないことがあるのです。それは，伸ばしてきた壁が自分自身の壁と交わらなければならない場合です。このようにトグロを巻いてしまったときは，その格子点から改めて作り直すことにしています。

このアルゴリズムは，比較的わかりやすい迷路ができやすいと思われます。

⦿X1バージョン

グラフィックラムが付いていることが条件です。110行で整数宣言をしていますが，これをしないとメモリが不足しますし，スピードも遅くなります。シャープ系BASICに慣れると忘れやすくなるので注意しましょう。RUNすると,乱数を聞いていきます。範囲内の数字を入力して CR または ENT を押してください。迷路作りの経過を見ることができます。

道の幅を1として，79×49という高密度なものになっています。作る途中の壁は白で，どこかの壁にぶつかると黄になります。ときどきトグロを巻いて消えていくところもおもしろいものです。

全部作り終わるには10分ほどかかります。この迷路をベースに,エネルギー,敵などを設定したり,立体表示に変えるなどアドベンチャーゲーム風の拡張を考えてみてください。

⦿MZ-2000, 80Bバージョン

MZ-2000, 80Bの共用プログラムです。ただし，両機種ともグラフィックラム1ページが必要です。

シャープ系BASICでは，1次元の配列を255までで切っているため，ここでは直接F000番地に座標を格しています。迷路の密度もX1バージョンとまったく同じです。

迷路は6分ほどで完成します。プリンタのある方はIMAGE命令でそのまま打ち出してもいいでしょう。

X1用高密度迷路

```
:00 REM  High Dense Maze for X1(with GRAM)
110 DEFINT A-Q,S-Z
120 DIM M(79,49),L(4000)
130 INIT:WIDTH 40:N1=78:N2=48
140 PRINT "INPUT RANDOM NUMBER(-32768=<X<=65535)":INPUT X:RANDOMIZE(X)
150 CLS:COLOR 2
160 FOR X=0 TO N1+1:M(X,0)=10000:M(X,N2+1)=10000:NEXT X
170 FOR Y=1 TO N2 : M(0,Y)=10000:M(N1+1,Y)=10000:NEXT Y
180 LINE(0,0)-(0,4*(N2+1))-(4*(N1+1),4*(N2+1))-(4*(N1+1),0)-(0,0)
190 LINE(0,0)-(4,0),PRESET:LINE(312,196)-(316,196),PRESET
200 FOR X=1 TO N1
210  FOR Y=1 TO N2
220   K=K+1
230   IF M(X,Y)>0 THEN 320
240   I=X:J=Y:M(I,J)=K:P=0:COLOR 7
250   L(P)=I:L(P+1)=J:P=P+2
260   IF (M(I+1,J)=K)AND(M(I,J+1)=K)AND(M(I-1,J)=K)AND(M(I,J-1)=K) THEN GOSUB 36
0:GOTO 240
270   R=INT(RND*4):I1=I+ABS(R-2)-1:J1=J-ABS(R-1)+1
280   IF M(I1,J1)=K THEN 270
290   LINE(I*4,J*4)-(I1*4,J1*4)
300   IF M(I1,J1)>0 THEN CLS 1:GOTO 320
310   I=I1:J=J1:M(I,J)=K:GOTO 250
320  NEXT Y
330 NEXT X
340 PLAY "C1DEFG3"
350 X$=INKEY$:IF X$="" THEN 350 ELSE END
360 FOR H=P-4 TO 0 STEP -2
370  LINE(L(H)*4,L(H+1)*4)-(L(H+2)*4,L(H+3)*4),PRESET:M(L(H),L(H+1))=0
380 NEXT H
390 M(L(P-2),L(P-1))=0
400 RETURN
```

MZ-2000、80B用高密度迷路

```
100 REM  High Dense Maze for MZ-2000,MZ-80B(with GRAM1)
110 LIMIT $EFFF:L=16*16*16*15
120 DIM M(79,49)
130 PRINT CHR$(6):GRAPH I1,C,O1:N1=78:N2=48
140 FOR X=0 TO N1+1:M(X,0)=0.5:M(X,N2+1)=0.5:NEXT X
150 FOR Y=1 TO N2:M(0,Y)=0.5:M(N1+1,Y)=0.5:NEXT Y
160 LINE 0,0,0,4*(N2+1),4*(N1+1),4*(N2+1),4*(N1+1),0,0,0
170 BLINE 0,0,4,0:BLINE 312,196,316,196
180 FOR X=1 TO N1
190 FOR Y=1 TO N2
200 K=K+1
210 IF M(X,Y)>0 THEN 300
220 I=X:J=Y:M(I,J)=K:P=0
230 POKE L+P,I:POKE L+P+1,J:P=P+2
240 IF(M(I+1,J)=K)*(M(I,J+1)=K)*(M(I-1,J)=K)*(M(I,J-1)=K)THEN GOSUB 350:GOTO 22
0
250 R=INT(RND(1)*4):I1=I+ABS(R-2)-1:J1=J-ABS(R-1)+1
260 IF M(I1,J1)=K THEN 250
270 LINE I*4,J*4,I1*4,J1*4
280 IF M(I1,J1)>0 THEN 300
290 I=I1:J=J1:M(I,J)=K:GOTO 230
300 NEXT Y
310 NEXT X
320 MUSIC "C1DEFG3"
330 GET X$:IF X$="" THEN 330
340 END
350 FOR H=P-4 TO 0 STEP -2
360 BLINE PEEK(L+H)*4,PEEK(L+H+1)*4,PEEK(L+H+2)*4,PEEK(L+H+3)*4:M(PEEK(L+H),PEE
K(L+H+1))=0
370 NEXT H
380 M(PEEK(L+P-2),PEEK(L+P-1))=0:RETURN
```

〔迷路の簡単な作り方〕

(1, 1)　(2, 1)　(1, 2)　(N1, N2)

(ア) 格子点を考えます。

(入口)　(イ)

(1, 1) の点は、上壁まで伸びました。

(ウ)

(1, 2), (1, 3)の点についても同様。

(2, 1)

(エ) (1, 4) の点はすでに壁が通ってるので(1, 5)へ。

(オ) このようにして、順にすべての点について壁を作れば、

入口　立派な迷路

(オ) 立派な迷路の出来上がり！

トグロの例

迷路例

作成中 I

作成中 II

E SOUND命令徹底活用の巻

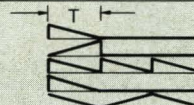
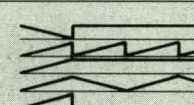
X1で音を出すとき，BEEP，PLAY，MUSIC命令などを用いますが，SOUND文が自由に使えるようになると，威力倍増といったところでしょう。ところが，マニュアルの説明は，かんたんな表が載っているだけであまりにも不親切なので，わかりやすく表にしてみました。活用してください。

このSOUND命令により，AY-3-8910という1チップシンセサイザのレジスタに，直接データが送り込まれます。このLSIはチャンネルを3つ持ち，アンプリチュード，エンベロープを設定することが可能で，さらにノイズを加えることもできるというものです。

表のR0からR5までは，それぞれのチャンネルの周波数，つまり音程を決めるものです。決定法の欄に出てくる2000000という数字は，CPUに送り込むクロックをさらに分周した周波数のことです。パラメータ値の(微)，(粗)というのは，それぞれ微調整，粗調整を表わします。したがって，大きく音程を変えたいときには(粗)の欄のレジスタの値を変えるようにしてください。

R6はサァーというノイズの音程を決めるものです。R8からR10でそれぞれのチャンネルの音程を決めます。0から15までの間の値にセットした場合，その音量の音が鳴り続けます(0では無音)。16とした場合はR11～R13で決定されるように音程が変化します。

エンベロープの周期を決めるのがR11とR12で，その形状はR13で決定されます。周期は図に示すような1単位のことです。

以上のR0からR13を変化させることで，ガンショット音，SL音などの効果音が自由に作れるのです。まず，一番かんたんなラの音を出してみましょう。

SOUND 1，1：SOUND 0，28 ↵
SOUND 7，62：SOUND 8，10 ↵

これがPLAY"A" ↵ による音の高さと同じことを確認してください。鳴り出した音は CTRL + D で消すことができます。プログラム中では，音量を0にすればよいと思います。

次にR6に31，R7に55，R8に16，R12に70，R13に0をセットしてください。ガンショットの音がリアルに出たはずです。レジスタの値を連続的に変化させたりして工夫してみてください。

ところで，エンベロープを用いるとき，すべてのレジスタに値をセットして音を出しているプログラムをよく見かけますが，MUSICやPLAY命令などを使わない限り，レジスタの値は保たれています。R13に値を入れるだけで音が出始めることも頭に入れておいてください。Elder Man氏の連載(82―88p.)参照してください。

SOUND命令活用法

レジスタ	内　　容	パラメータ値	パラメータ決定法
R0	チャンネルAの周波数	0←高(微)　低→255	出したい周波数をf_0ヘルツとする時 $S=\frac{2000000}{16\times f_0}=R_H\times256+R_L$ つまり $\begin{cases}R_H=INT(S/256)\\R_L=S-256\times R_H\end{cases}$ $\begin{pmatrix}R_H=R1, R3, R5\\R_L=R0, R2, R4\end{pmatrix}$
R1		0←高(粗)　低→15	
R2	チャンネルBの周波数	0←高(微)　低→255	
R3		0←高(粗)　低→15	
R4	チャンネルCの周波数	0←高(微)　低→255	
R5		0←高(粗)　低→15	
R6	ノイズ周波数	0←高　低→33	出したい周波数をf_1ヘルツとする時 $R6=\frac{2000000}{16\times f_1}$
R7	チャンネル設定	0　～(☞)　16	チャンネルA：トーン→－1　ノイズ→－8 チャンネルB：トーン→－2　ノイズ→－16 チャンネルC：トーン→－4　ノイズ→－32 $R7=63+\sum X$ ㊎チャンネルBからトーンとノイズを出したい時 R7=63－2－16=45
R8	チャンネルA音量	0←小　大→15	16にすると，音量は，R11～R13で決定されるように変化する。
R9	チャンネルB音量	0←小　大→15	
R10	チャンネルC音量	0←小　大→15	
R11	エンベロープ周期	0←小(微)　大→255	周期をT秒(R13の項参照)とする時 $U=\frac{2000000\times T}{256}=R12\times256+R11$ つまり $\begin{cases}R12=INT(U/256)\\R11=U-256\times R12\end{cases}$
R12		0←小(粗)　大→255	
R13	エンベロープ形状	0　～　15	T 0～3　4～7　8　9　10 11　12　13　14　15 音量　時間

F X1先進グラフィック機能活用虎の巻

X1のグラフィックは他機種を圧倒するほどの機能を持っているということに異論はないと思います。反面内容が濃いだけにちょっといじったくらいでは理解できないとも言えます。頼みのマニュアルを1回読んだぐらいではちょっと……という感じです。よく読むと，逆に疑問なところさえあるような気がしてきます。

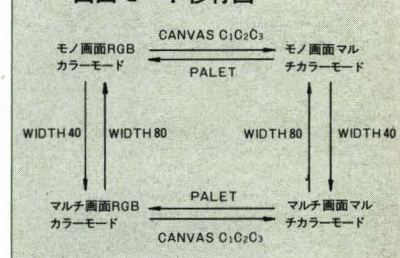
たとえば，1ページの特長紹介のところで，320×200ドットマルチ画面という言葉がでてきます。このマルチ画面という言葉は102ページの図の中にも出てきますが，明らかに両者での使われ方の意味するところが違います。

そこで，グラフィック画面やモードなどの意味をわかりやすくまとめてみます。X1ユーザーの方のお役に立てれば幸いです。

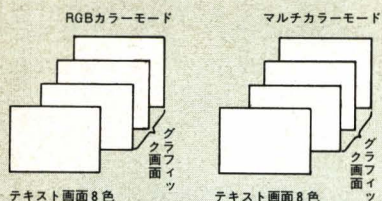
画面モード分類

A．モノ画面モード｛(80×25キャラクタ/640×200ドット) 1枚｝
　i) RGBカラーモード
　ii) マルチカラーモード
B．マルチ画面モード｛(40×25キャラクタ／320×200ドット) 2枚｝
　i) RGBカラーモード
　ii) マルチカラーモード

画面が3枚，6枚などというやり方もありますが，独立に持てるのはBのマルチ画面モードのときだけです。この場合，画面に表示されたカラーグラフィックとは別にプログラムでもう1画面グラフィックを作っておいて，瞬時に切り換えることができます。それぞれ8色のカラーを表示することができます。

画面モード移行図

画面モードは上図に示したコマンドにより移行します。PALETと書かれているのは，パラメータを付けずに実行することを意味しているのでご注意を。

初期状態ではマルチ画面RGBカラーモードです。

次に画面の表示され方を図に示します。

グラフィック画面(G1、G2、G3)間には優劣がなく、重ね合わせで8色を実現する。テキスト画面とグラフィック画面の優劣はPRW命令で定める。
PALET命令で表示カラーを変更する。

グラフィック画面の色はCANVASC1,C2,C3で指定する。各グラフィック画面及びテキスト画面の優劣はLAYER命令で定める。CANVAS命令でPALETを変えているのでPALET命令は使わない方がよい。

2つのモードの差を理解することができたら，ひとつの山を越えたと言えるでしょう。

カラーグラフィックパターンの表示法

①PSGにキャラクタとして定義しテキスト画面に表示②PATTERN文でグラフィック画面に表示③GET@、PUT@命令の活用。

それぞれの場合については，実際にやってみるのが一番の近道です。直接G-RAM，V-RAMに書き込む命令OUT，POKE@もありますが，最初は使わないほうがよいでしょう。参考のためI/Oマップを示します(主記憶空間から分離されているのに注意)。

I/Oマップ

アドレス	内容
0000	システムI/O
2000	文字アトリビュート
3000	文字V-RAM
4000	G-RAM BLUE
8000	G-RAM RED
C000	G-RAM GREEN
FFFF	

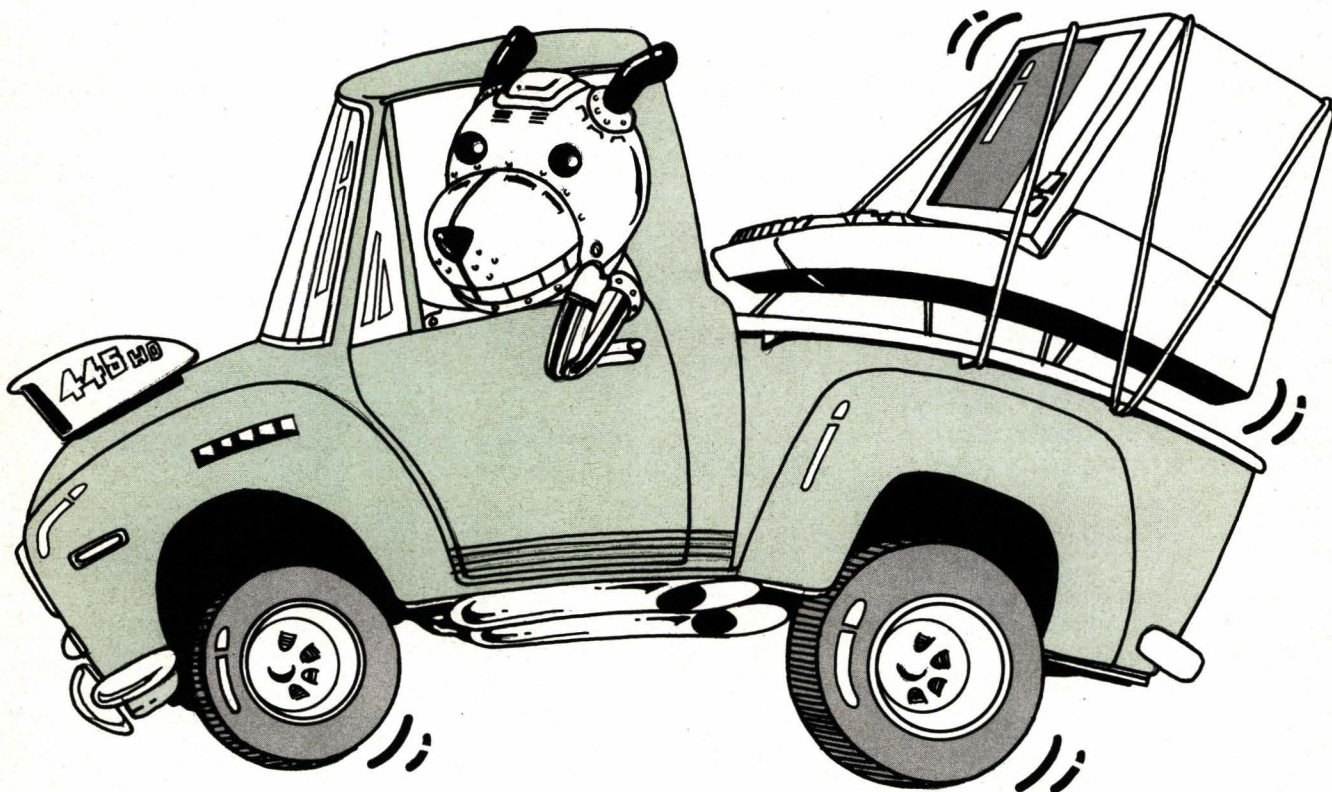
PSET，LINE，CIRCLE，POLY命令で図形を書く際，色の指定が2つのモードで違うので注意してください。

グラフィック図形の色の指定

i）RGBカラーモード(8色)
　パレットコードで指定する。省略の時は、GRAPH(SCREEN) により指定されたモード。ただしグラフィックモード0を指定した場合は、COLOR文で指定されたカラー。
ii）マルチカラーモード(8色中3色)
　SCREEN(GRAPH)により指定されたグラフィック画面のカラー。

WINDOWの指定に関しては，マニュアルに詳しく書かれているので省略します。

話はちょっと横道にそれますが，グラフィックラムを外部記憶用に使うのも興味深いので付け加えます。

外部記憶メモリとして使うには，OPTION，SCREEN 2を実行すればよいのです。カセットのプログラムを，読み込んではG-RAMにSAVEすることを繰り返せば，そのあとはディスクよりアクセスの遠いものになります。もちろんFILES⏎などもできます。ただし，グラフィックラムを使うプログラムの実行はできませんし，作り直したプログラムは改めてカセットにセーブしておかないと，パーになります。

MZ-80B/B2/2000/2200

HuBASIC Ver 2.0 徹底紹介

松崎　洋

はじめに

パソコンのほとんどが，BASICインタプリタをROMで本体内部に装備しています。確かに，初心者にとって，BASIC言語というのは理解しやすく，処理速度，メモリ容量などの問題は特に問題視しなくとも充分使いこなせることは事実です。エンドユーザー（この場合，ホビーユース）や実務用途に使用される場合はこれで充分ですが，ROM-BASIC以外の言語を運用するとなるとたいへんです。フロッピーベースのスタンダードOS（CP/M-80など）は現在ほとんどのパソコンで利用できますが，メモリマップ上にROMがあった場合，非常に大きな問題が生じてしまいます。仮に，CP/M-80などを起動することにより，これらのOSの管理下となるため，ROMがじゃまになってしまうのです。

本来，コンピュータというものは，OSに属するソフトウェアは起動時に各種媒体からロードするのが普通ですが，パソコンはどうやら違うようで，今後もこの傾向は続くものと思われます。

MZシリーズの上位タイプ，80B/2000/3500は，CPUが直接管理するメモリをすべてRAMとしている，いわば本来のコンピュータであるといえます。メーカーは「クリーン設計」とPRしていますが，メインメモリオールRAMの利点はソフトサポートが完璧であれば最高の環境なのです。

このメインメモリオールRAMを駆使して内外のソフトメーカーから多種のランゲージプロセッサがリリースされていますが，今回新たにハドソンからMZ-80B／2000用に同社オリジナル「HuBASIC」が完成(MZ-2000用については既報)，発売されました。この完成により，X1およびMZシリーズ(3500を除く）のすべてに「HuBASIC」が出揃いました。

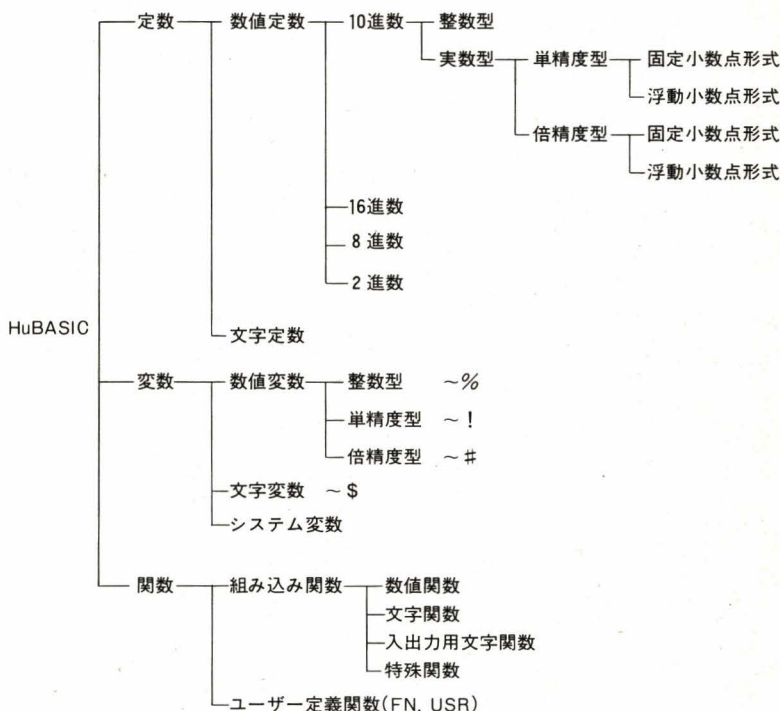
本稿ではこの「HuBASIC」の概要を詳細に紹介します。先ほども少し触れましたが，本誌前々号で既報していますので，支障のない限り重複する部分は割愛いたしました。ご了解ください。

HuBASICの概要

今回リリースされた「HuBASIC」は，一昨年MZ-80B用として発表された「Hu-GBASIC/80B」の事実上のバージョンアップであり，CZ-800Cの「Sharp-HuBASIC」およびMZ-700の「HuBASIC」との互換性（後述）をある程度持つVersion 2.0です。命令体系は，なぜか他社が良く採用する米国マイクロソフト社製BASICに近いため，比較的扱いやすいと思いますが，シャープBASIC（SB系，MZ-xZ系など）しか使ったことのないユーザーの方にとっては若干手間取るかも知れません。

特長1　200を超える豊富な命令群

HuBASICの特長のひとつ目として，コマンドが豊富であることが挙げられます。このV2.0，約210種の命令群を持っています。MZ-2000に付属するMZ-1Z001が約95といわれていますから，2倍以上のコマンドがあるわけです。実際にはすべてを自在に使いこなすにはかなりの時間がかかるでしょう。しかし，今までサブルーチンを作り，処理していたものが，命令ひとつで済むようになるのですから便利といえば便利です。また，210種を超える命令群であるため，比較的使用頻度の高いコマンドに関しては省略形入力ができるディファイナブルファンクションキーのような手軽さで扱えるというのも喜ばしいことです。

図1　HuBASIC　V2.0

特長2 構造化プログラミングが可能

一般的に，BASICと呼ばれるインタプリタ言語は，さほど処理速度を気にしなければ，CRTコーディングなどといわれるように，即席で考えながらキーボードを叩いてもそれなりの処理ができます。しかし，対話型以外の言語（Fortran, Cobolなど）はいわゆるスパゲッティはできません。そこで構造化が不可欠となるのです。HuBASICは，FOR～NEXTの他，REPEAT～UNTIL, WHILE～WENDのループ処理，そしてLABELが付けられるなど構造化プログラミングに便利な命令が用意されています。

特長3 グラフィック処理機能の強化

最近のパソコンは，グラフィックをセールスポイントとしているものが増えつつあります。MZ-80B/2000は，一応のグラフィック対応機ですが，シャープが供給しているソフトウェアでは，このグラフィックを十二分に活用することはできませんでしたが，HuBASICではかなり強化されており，使いやすそうです。まず，座標X，Y軸とも±1E38の範囲内でスケールも自在に設定できます。これにより，算出した変数を一度座標変換する手間が軽減できます。そして，タイリングペイントで中間色や自由なパターンでぬりつぶしができたり，スタイル指定によるLINE文があり，実線の他点線，破線，1点鎖線なども簡単に出力できます。

特長4 演算機能の強化

パーソナルといえど，コンピュータですから，やはり演算機能は重視したいものです。シャープ系のBASICはフリーエリアを大切にしているため，実務計算を行なうには別に倍精度BASICを求める必要があります。HuBASICでは，図1に示すとおり各種の精度が使えます。

〈単精度型〉

有効桁数が8桁以内で，約-1.7×10^{38}～1.7×10^{38}までの数を表すことができます。桁数が8桁以内の場合は固定小数点形式で表示され，8桁を超えると浮動小数点形式で表示されます。

図2 ファイルディスクリプタと入出力デバイス

ファイルディスクリプタ	入出力デバイス名	用途		ファイルモード指定	
		入力	出力	A	R
KEY：	キーボード	○	×	×	×
CRT：	ディスプレイ	×	○	×	×
SCR：	ディスプレイ	×	○	×	×
LPT：	プリンタ	×	○	×	×
CAS：	カセットデッキ	○	○	×	×
0： ≀ 3： 4： ≀ 7：	ディスクドライブ1 ≀ ディスクドライブ4 ディスクドライブ1 ≀ ディスクドライブ4	○	○	○	○
MEM：	グラフィックメモリ	○	○	○	○
EMM0： ≀ EMM9：	外部メモリ0 ≀ 外部メモリ9	○	○	○	○

〈倍精度型〉

有効桁数16桁以内で，取り扱える数の範囲は単精度型と同じです。倍精度の場合，16桁の精度でメモリに格納され，表示も16桁で行ないます。この倍精度型による関数演算も可能です。

また，演算に不可欠な変数名もロングバリアブルネームの採用により240文字まで使え，かつアップバーや英小文字も変数に使えますから，構造化プログラミングに適した命令や字下げを有効に使うことによりプログラムの保守性を向上させることができます。

特長5 しっかりしたファイル概念

コンピュータは，各種のデバイス（周辺機器）からひとつのシステムを形成しています。MZ-80B/2000は，オールインワン設計ですが，正確にはこの本体の中にキーボード，カセットデッキ，CRTのデバイスが一体になっているといえます。まして，システムとして拡張しますと，プリンタやらフロッピーなどという機器類が本体内CPUの制御を受けるわけですから，しっかりとしたファイル概念が不可欠です。たとえば，現在メモリ上にあるプログラムリストを出力デバイス（プリンタ）に出力する場合，インタプリタレベルからでは次のようになります。

①LIST/P CR　　シャープBASIC
②LLIST CR
③LIST "LPT：" CR

いずれも同じ働きをしますが，ファイル概念から考えた場合③がもっとも正当であるといえます。すなわち，リストをデバイス名"LPT："に出力せよといった意味です。M-BASICやHuBASICでは②，③のいずれでも可能ですが，どちらかというと②は便宜上であるといえそうです。

このように，ファイル概念は大切なことです。ここで図2にHuBASICのファイルディスクリプタと入出力デバイスの関係を示します。この図からもわかるとおり，HuBASIC Ver2.0ではファイル概念が良くまとまっていると思います。

特長6 キャッシュディスク対応

前項でも触れましたが，このファイル概念により，外部メモリを利用してデータを格納することができます。使用できるメモリはグラフィック用RAM（MEM：）およ

図4

●プログラム（1） <FOR-NEXTループ>

```
10 TIME$="00:00:00"
20 FOR I=1 TO 10000
30 NEXT I
40 PRINT TIME$
50 END
```

●プログラム（2） <画面処理速度の測定 80文字モード>

```
10 TIME$="00:00:00"
20 FOR I=1 TO 1000
30 PRINT I
40 NEXT I
50 PRINT TIME$
60 END
```

●プログラム（3） <加算速度の測定>

```
10 TIME$="00:00:00"
20 A=1
30 A=A+1
40 IF A<1000 THEN 30
50 PRINT TIME$
60 END
```

●プログラム（4） <四則演算関係>

```
10 TIME$="00:00:00"
20 A=1
30 H=A+1
40 I=A-H
50 J=A*I
60 K=A/J
70 IF A<1000 THEN A=A+1:GOTO 30
80 PRINT TIME$
90 END
```

●プログラム（5） <乱数作成速度>

```
10 TIME$="00:00:00"
20 FOR I=1 TO 1000
30 A=RND(1)
40 NEXT I
50 PRINT TIME$
60 END
```

●プログラム（6） <データ読み込み速度>

```
10 TIME$="00:00:00"
20 DIM A(10)
30 FOR J=1 TO 100
40 RESTORE
50 FOR I=1 TO 10
60 READ A(I)
70 NEXT I
80 NEXT J
90 PRINT TIME$
100 END
110 DATA 1,2,3,4,5,6,7,8,9,0
```

●プログラム（7） <GOTO文処理速度>

```
10 TIME$="00:00:00"
20 FOR I=1 TO 10000
30 GOTO 40
40 GOTO 50
50 GOTO 60
60 GOTO 70
70 NEXT I
80 PRINT TIME$
90 END
```

●プログラム（8） <GOSUB文処理速度>

```
10 TIME$="00:00:00"
20 FOR I=1 TO 10000
30 GOSUB 100
40 GOSUB 100
50 GOSUB 100
60 GOSUB 100
70 NEXT I
80 PRINT TIME$
90 END
100 RETURN
```

●プログラム（9） <文字列操作>

```
10 TIME$="00:00:00"
20 A$="ABCDEFGHIJ"
30 B$="KLMNOPQRST"
40 FOR I=1 TO 1000
50 C$=A$+B$
60 C$=A$
70 C$=""
80 C$=B$
90 NEXT I
100 PRINT TIME$
110 END
```

●プログラム（10） <三角関数関係処理>

```
10 TIME$="00:00:00"
20 FOR J=1 TO 100
30 FOR I=1 TO 10
40 X=SIN(I)
50 X=COS(I)
60 X=TAN(I)
70 NEXT I
80 NEXT J
90 PRINT TIME$
100 END
```

び320Kbytes大容量RAMボード（EMM：*(注)）で，とりわけ後者のRAMボードは，フロッピーディスクの代用として用いれば，メモリですからディスクのハードに依存するアクセスよりはるかに早く入出力が可能となります。使用例として日本語ワープロを行なう場合，いちいち辞書をFDから引かなくとも，一時内容をEMM：に転送し，オンメモリの状態で文章作成を行ない，終了後FDにデータとともに書き込むことにより効率が向上すると思います。

＊(注) PIO-3034 320キロバイト大容量
RAMボード ￥98,000
発売元 ㈱I・Oデータ機器

特長7 漢字出力機能

同じく㈱I・Oデータ機器製漢字ROMボード（PIO-3055）を装備することにより，簡単に漢字を出力することができます。カナ漢字変換機能をサポートしているため，ユーザーレベルで簡単にワードプロセッサが作成できます。

特長8 異機種間の互換性

シャープBASIC管理下であれば80B，2000相互に記録媒体から読むことができますが，このHuBASIC V2.0の場合，80B，2000はもとよりボーレートをモニタレベルで変更することにより，CZ-800C，MZ-700のHuBASIC管理下の各種ソフトウェアを読み込むことができます（アスキー，バイナリ形式とも)。ただし，各機種特有の命令，機能はサポートされていません。また，HuGBASIC上で作成したソフトウェアはそのままでは記録フォーマットが異るため読めませんが，付属のコンバートプログラムにより対処できます（MZ-2000用にも付属しています)。

図3 モニタのコマンド

＊P (Printer switch)
＊D (Dump)
＊M (Memory set)
＊F (Find)
＊G (Gosub)
＊T (Transfer)
＊S (Save)
＊L (Load)
＊V (Verify)
＊R (Return)
＊B ボーレート変更

デフォルト値 (1200)

ボーレート	LOAD/Verify	SAVE
1200	＊BL 1	＊BS 1
2000	＊BL 2	＊BS 2
2700	＊BL 3	＊BS 3
2000 ↓ 1200	＊BL＋	＊BS＋
2000 ↓ 2700	＊BL−	＊BS−

図5 ベンチマークテスト測定結果

プログラム番号	HuBASIC	SB-6520	SB-6620	MBASIC
1	00:08	00:06	00:07	00:15
2	00:27	00:26	00:24	00:30
3	00:04	00:05	00:05	00:04
4	00:15	00:12	00:16	00:14
5	00:03	00:04	00:07	00:04
6	00:07	00:05	00:06	00:05
7	00:16	00:37	00:37	00:45
8	00:33	01:03	01:04	01:14
9	00:09	00:09	00:10	00:08
10	01:20	00:43	——	00:50

<使用したソフトウェア>
1. HuBASIC V2.0/FD (W-3650-D) ハドソン・ソフト
2. ディスクBASIC (MZ-8BDM) シャープ
3. 倍精度ディスクBASIC (MZ-8BD01) シャープ
4. BASIC-80 REV 5.22 [CP/M Version]
Created:10-MAR-82 (C)Microsoft

処理速度，メモリ構成

では，HuBASIC V2.0はどれだけすぐれているのでしょうか。単に，ソフトウェアはひとことで優劣の判断は下せないということは読者の皆さんも良くおわかりのことと思いますので，この項ではちょっとした実験を試みることにします。ベンチマークテストを行なうわけですが，ソフトウェアのみを評価する上で同一の機種を使用しないとまったく意味がなくなってしまいますのでここでは私が所有しているMZ-80Bを利用することにします。比較対象としたソフトウェアは次の3本です。

①SB-6520（MZ-2Z001相当）
②SB-6620（MZ-2Z003相当）
③BASIC-80（マイクロソフト社）

❖

①は何のことはないマスターディスケットと呼ばれるもの，②は内部10進型の倍精度ディスクBASIC（以上シャープ製），そして③はCP/M-80上で動作するいわゆる高級言語のひとつです。OSはMSAが供給しているVer 2.2Aを使用しました。

ベンチマークの内容，結果は，表にまとめましたので参照してください。

HuBASICはかなり巨大なBASICですから，処理速度に関しても相当遅いのではないかという予想がでてきますが，結構好結果が出ています。特に驚いたのが，GOTO，GOSUBの処理の速度はいずれも2分の1の速さを示しており，最適化がかなり良くなされているものと思われます。

❖

図6にHuBASICのメモリマップを示します。80Bのユーザーの方で以前Hu-GBASICを使ったことのある方は首をかしげるかも知れませんが，大幅に異なっているようです。以前（Hu-GBASIC）は，Hu-Monitorと呼ばれるIOCSを含むモニタとインタプリタの2つにより形成していました（おおよそシャープBASICにおける方式と同じ）が，V 2.0では日電のPCシリーズのようなメモリ構成となっています。単に，インタ

プリタレベルではバージョンアップとの表現で済みますが，どうやら作り直したといったほうが正しいものと思われます。また，メモリマップが大幅に異なるため，Hu-G BASIC上で作成したオブジェクトファイルのV 2.0での動作は簡単にいかないかもしれません。

図7にV 2.0における中間コードを示します。V 2.0では，FDバージョンとカセットバージョンでは完全なコンパチビリティがあります。テキストエリアはA4BO〈H〉付近からのようですので*格納状況を一度調べてみるのも良いでしょう。本稿で示した中間コードは，筆者が解析したものであり，発表するにあたり充分チェックしましたが，不都合がある場合はご指摘いただければ幸いです。

＊MZ-80B用フロッピーバージョンの場合

80B/2000の相異点

MZ-80BとMZ-2000はハードウェア上の違いにより，HuBASIC本体の各グラフィック命令に相違が生じます(図8参照)。これに伴い，COLOR, CLS, WINDOW, PAINT, POSITION, HCOPYそしてPALETの各命令に関して制約があります。主にカラー関係であるため，無視してもかまいませんが，カラーを使う場合，I・Oデータ機器製のグラフィックカラーボードを使用することにより対処できます。

評価

HuBASIC V2.0は，MZ-700およびCZ-800CにHuBASICが標準装備されたことにより，かなりシャープ系のBASIC (SB系, MZ-1Z系) のコマンドを取り入れています。たとえばFREに対しSIZEも使えたりというように，一般的に独特のシャープBASICしか知らないユーザーへの配慮からだと思われます。

他に長所として，スクリーンエディットの強化があります。シャープBASICでは良く行割れや2行分が1行になったりすることがありましたが，HuBASICではまず起こりません。また，プログラムは小文字で入力することが許されます（ただし，変数名はそのままとなる)。コントロールコードの装備も喜ばしいことです。

逆に，不満点も数点あるようです。第一にユーザーフリーエリアが約20Kバイトと少ないこと。システムが巨大になればフリーエリアが減るのが当然ではありますが，それにしても一番気になることではないでしょうか。BREAK処理については，コントロールコードで送るか[SHIFT]＋[BREAK]を同時に送る必要があります。今までは，[BREAK]キーのみでしたが，このキー単独では[ESC]キーのような働きとなってしまいますので戸惑うかもしれません。操作性に関しても，ハードウェアにそった工夫が欲しかったと思います。BREAK処理はキー入力後ワンテンポ処理が遅いようです。

すでにHuBASIC V 2.0をお持ちの方たはちょっと試してください。ブートしたら，カナモードにしてRESETボタンを押してみてください。カーソルはカナモード表示なのにキーは英文字が入力されます。ちょっと不可解な現象ですが，のちのち改良されるでしょう。

HuBASIC管理下では，異機種間の互換性があると先ほど書きましたが，正確にはボーレートが変更できる機能を持っている

図6 メモリマップ

アドレス	領域	説明
0000	IOCS	←&H0000
	モニタ	
	BASIC インタープリタ	
	テキスト	←NEW ONで指定したテキストアドレス
	変数エリア	→テキストエンドコード "00 00"
	ファイル用ストリングバッファ ＃0，＃1，…	→変数エンドコード "00" MAXFILESnでとられるエリア (n+1)×(16+256)バイト
	ストリングデータバッファ	
	テンポラリーストリングエリア	
	↓ フリーエリア ↑	
	STACK	
	FAC	←(CLEAR, LIMIT)−&H100
	ユーザー用機械語フリーエリア	←CLEAR, LIMITで指定したアドレス
FF00	モニタ用ワークエリア	←&HFF00
FFFF		←&HFFFF

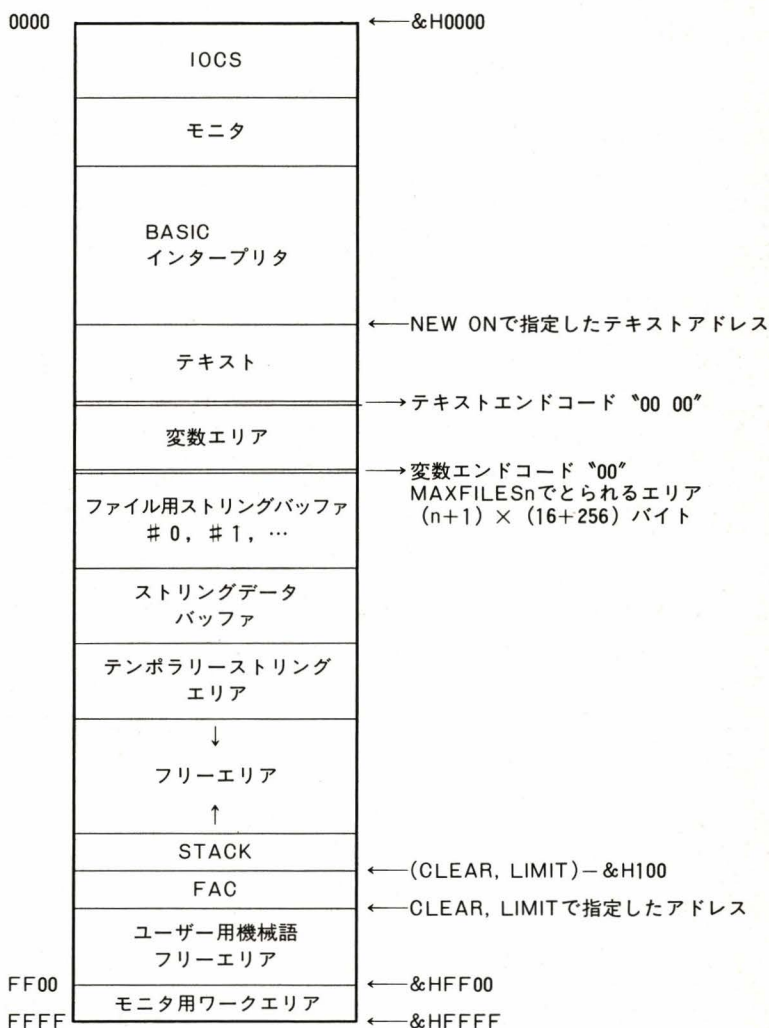



図7 **中間コード表**

コード	命令	コード	命令	コード	命令	コード	命令
80	GOTO	A0	POKE	FE80	WINDOW	FEA0	
81	GOSUB	A1	OFF	FE81	PSET	FEA1	GRAPH
82	GO	A2	WHILE	FE82	PRESET	FEA2	MUSIC
83	RUN	A3	WEND	FE83	COLOR	FEA3	TEMPO
84	RETURN	A4	REPEAT	FE84	CIRCLE	FEA4	CURSOR
85	RESTORE	A5	UNTIL	FE85	POLY	FEA5	VERIFY
86	RESUME	A6		FE86	PAINT	FEA6	CLR
87	LIST	A7		FE87		FEA7	LIMIT
88	LLIST	A8		FE88	POSITION	FEA8	KLIST
89	DELETE	A9	TRON	FE89	PATTERN	FEA9	
8A	RENUM	AA	TROFF	FE8A	HCOPY	FEAA	
8B	AUTO	AB		FE8B	PLAY	FEAB	CLICK
8C	EDIT	AC		FE8C		FEAC	BOOT
8D	FOR	AD		FE8D	BEEP	FEAD	DEVI$
8E	NEXT	AE	DEFINT	FE8E		FEAE	DEVO$
8F	PRINT	AF	DEFSNG	FE8F	PALET	FEAF	MOTOR
90	LPRINT	B0	DEFDBL	FE90		FF80	INT
91	INPUT	B1	DEFSTR	FE91		FF81	ABS
92	LINPUT	B2	DEF	FE92		FF82	SIN
93	IF	B3		FE93		FF83	COS
94	DATA	B4	LOAD	FE94		FF84	TAN
95	READ	B5	SAVE	FE95		FF85	LOG
96	DIM	B6	MERGE	FE96	EJECT	FF86	EXP
97	REM	B7	CHAIN	FE97	CSTOP	FF87	SQR
98	END	B8	CONSOLE	FE98	FAST	FF88	RND
99	STOP	B9	WIDTH	FE99	REW	FF89	PEEK
9A	CONT	BA	OUT	FE9A	APSS	FF8A	ATN
9B	CLS	BB	SEARCH	FE9B		FF8B	SGN
9C	CLEAR	BC	WAIT	FE9C		FF8C	FRAC
9D	ON	BD	PAUSE	FE9D		FF8D	FIX
9E	LET	BE	WRITE	FE9E		FF8E	PAI
9F	NEW	BF	SWAP	FE9F		FF8F	RAD
C0		E0	TO	FF90	INP	FFB0	CVI
C1	ERROR	E1	STEP	FF91	CDBL	FFB1	DEVF
C2	ELSE	E2	THEN	FF92	CSNG	FFB2	
C3	CALL	E3	USING	FF93	CINT	FFB3	ERR
C4	MON	E4	SUB	FF94		FFB4	ERL
C5	LOCATE	E5	BASE	FF95	EOF	FFB5	CSRLIN
C6	SCREEN	E6	TAB	FF96	FPOS	FFB6	STRPTR
C7	KEY	E7	SPC	FF97	LOC	FFB7	DTL
C8		E8	EQV	FF98	LOF	FFB8	
C9		E9	IMP	FF99	POS	FFB9	
CA	LABEL	EA	XOR	FF9A	FAC	FFBA	LEFT$
CB	RANDOMIZE	EB	OR	FF9B	SUM	FFBB	RIGHT$
CC	OPTION	EC	AND	FF9C	FRE	FFBC	MID$
CD	LINE	ED	NOT	FF9D	LPOS	FFBD	INKEY$
CE	OPEN	EE	><	FF9E	STICK	FFBE	INSTR
CF	CLOSE	EF	<>	FF9F	STRIG	FFBF	HEXCHR$
D0	SIZE	F0	=<	FFA0	CHR$	FFC0	MEM$
D1	FIELD	F1	<=	FFA1	STR$	FFC1	SCRN$
D2	GET	F2	=>	FFA2	HEX$	FFC2	VARPTR
D3	PUT	F3	>=	FFA3	OCT$	FFC3	STRING$
D4	SET	F4	=	FFA4	BIN$	FFC4	TIME
D5	FILES	F5	>	FFA5	MKI$	FFC5	
D6	LFILES	F6	<	FFA6	MKS$	FFC6	
D7	DEVICE	F7	+	FFA7	MKD$	FFC7	FN
D8	NAME	F8	−	FFA8	SPACE$	FFC8	USR
D9	KILL	F9	MOD	FFA9		FFC9	CALC
DA	LSET	FA	¥	FFAA	KANJI$	FFCA	
DB	RSET	FB	/	FFAB	ASC	FFCB	ATTR$
DC	INIT	FC	*	FFAC	LEN	FFCC	POINT
DD		FD	^	FFAD	VAL	FFCD	CHARACTER$
DE	MAXFILES	FE		FFAE	CVS	FFCE	CMT
DF		FF		FFAF	CVD	FFCF	MIRROR$
						FFD0	JIS

ため，テープが読めるということです。単にバイナリー形式で記録したものも読めますが，予約語テーブル上にない命令（各機種特有のコマンド）がある場合，正確なプログラムとしてロードできませんので，アスキー形式でセーブするのが好ましいようです。このボーレート変更はモニタでBコマンドにより行ないますが，できればBASIC上でファイルディスクリプタ等の拡張により処理できたほうが好ましいように思えます。また，これだけ大きなシステムに対してシリアル(RS-232C)サポートがないのがとても残念です。

以上，総括的に良くできたBASICインタプリタといえます。細かな点での不満が若干ありますが，処理速度，機能といい申し分ないものです。

図8 **MZ-80B用の相違点**

①SCREEN命令　表示モード　0～15→0～3　　アクセスモード　0～3→0～2

②COLOR　　表示色　0～7→0～3

③CLS命令　　n=0, 1, 2, 3, 4→n=0, 1, 2, 4

④PALET命令

文法　PALET〔パレットコード〕〔，カラーコード〕
　　パレットコード　0～7　　カラーコード　0～7

説明　I/Oデータ機器より発売しているカラーボードを使用すると，カラーディスプレイと接続して，カラー表示が可能となります。その色を決定するための命令です。

パレットコード	グラフィック及びテキストの状態	カラーコード	色
0	何もセットされていない	0	黒
1	グラフィック1のみセット	1	青
2	グラフィック2のみセット	2	赤
3	グラフィック1および2セット	3	マゼンダ
4	テキストのみセット	4	緑
5	テキストとG1セット	5	シアン
6	テキストとG2セット	6	黄色
7	テキストとG1とG2セット	7	白

左のパレットコードを右の任意のカラーコードに指定できます。

⑤WINDOW命令　Xs及びXe　0～639→0～319

⑥PAINT命令　タイルパターン　3文字→2文字

⑦POSITION　X水平座標　0～639→0～319

⑧HCOPY　1～3→1～2

MZ-2000

実感 16ビット体験

風間　浩

近ごろよく,「16ビットパソコン」という言葉を聞きます。私たちMZ ユーザーにとってはあまり関係なさそうだ，とはいっても，多少なりとも気になってしまう……。こんな人も多いのではないでしょうか。

というところへ，MZ-2000で16ビットと関係ができる(何のこっちゃ?)キットが，シャープから発売されました。それも，ただの「関係」ではなく，MZ-2000が16ビットマシンそのもの! に変わってしまうという，かなりすごそうなヤツです。ほんとうにすごいかどうか，また，これまでとどう変わるのか，さっそく使ってみましたので，詳細を紹介していきたいと思います。

16ビットと8ビット，どこが違うのか

16ビットは8ビットの2倍ダ! と言われていますが，要するに16ビットのデータが同時に扱えるということです。CPUでの演算が，Z-80では8ビット単位で，16ビットCPUでは16ビット単位で，ということになっています。Z-80にはADD HL, DEのような命令があって，16ビットのデータを一度に操作できるのですが，これはほんの一部に限られていて，16ビットCPUと呼ぶほどではありません。また，一般の16ビットCPUは，メモリからCPUにデータを読み出す（書き込む）ための信号線，**データバス**が16本あり，8ビットCPUの8本と比べ，原理的には倍のスピードでプログラムの実行ができるわけです。すなわち，今のアドレスと次のアドレスの値とを一度に読み出せると考えられます。

それから，たくさんのメモリをつなぐことができるよう，**アドレスバス**が16本から20本に拡大され，64Kバイトのメモリ空間が1Mバイト(1024バイト)になりました。これで，バンク切り換えの手法を用いなくても，数多くのメモリを利用することができます。MZ-2000ではメインメモリの64K以外に，キャラクタV-RAMが2K，オプションですがグラフィックRAMが48Kほどを扱っていますから，バンク切り換えをしていますね。ただ，アドレス空間が増えたことで，若干の制限も出てきますが，十分に帳消しできる範囲です。

こんなところが，一般的な「16ビットCPU（システム)」の特徴です。

8ビットCPUには，圧倒的シェアをもつZ-80のほかにも，6809，6800，6502，8080，8085などがありましたが，16ビットCPUにもいくつかの種類があります。インテル・NEC・各社系で8080・8085の上位となる「8086・8088」，ザイログ・シャープ・東芝系でZ-80の上位となる「Z-8000」，モトローラ・日立系で6800・6809の上位となる「68000」と，だいたい3系統にまとまっています。

性能面では，8088<8086<Z-8000<68000という評価が一般的(?)ですが，普及率では逆に，8088・8086がトップで，国内のほとんどの「16ビットパソコン」で採用されています。制御用にはZ-8000が使用されているのを見かけたりもしますが，パソコンではまだのようです。68000は，一部のパソコンでは使用されています。

話は少し難しくなりますが，8086と8088の違いについて話しておきましょう。8086も8088ももちろん16ビットCPUですが，データバスが8086は16本，8088は8本です。他のところは，レジスタや命令そのものについてもまったく同じです。したがって，8088は16ビットCPUのもつスピードアップの効率が薄れてくるのですが，ハード構成でみると「8ビットCPUの置き換え」として使えるところにメリットがあるのです。つまり，メモリに関する部品点数が半分(8/16)ですむ（要するに8ビット機と何ら変わりがない）わけで，コストダウンになります。もちろん「16ビットCPU」を採用した「16ビットパソコン」には違いありませんし，8086ソフトコンパチブルと言えますから，イメージアップにもつながります。国内はもちろんのこと，IBMの16ビットパソコンでさえ「8088」を採用していることが多いのは，このへんに理由があるようです。

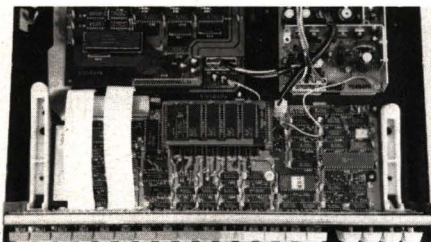
MZ-2000 16ビットボード

16ビットに関する状況がわかったところで，そろそろ本論に入りましょう。

MZ-2000，16ビットボードキットは，CPUに「8088」を採用し，IPL ROMが8Kバイト（うち，約2Kバイトのみ使用)，RAMが128Kバイト（今までの倍）のクリーン設計になっています。CPUに自社製のZ-8000を使わなかったのは,「8088ならコストダウンできる」,「ソフト的にもっとも8ビットに近いため，BASICインタープリタを作りやすい」,「他社に合わせたかった」などの理由があるからと思われます。これも，8085のメーカーがZ-80（コンパチ）を使ったパソコンを出していることを考えれば，特別気にする必要もないでしょう。また，8086でなく，8088にした理由は前述のとおりで，ここで言うまでもありません。

CPU，メモリの他に，「漢字ROMソケット」も付いています。つまり，漢字ROMは128Kバイトですから，16ビットCPU（アドレス空間は1Mバイト!）にかんたんに付く，という特典を生かしたものです。この「漢字ROM」は割安で，¥29,000です。

16ビットボードキットには，この他に「16ビットBASIC　1Z-012」と「アプリケーション（プログラム3本入り)」も付いてきます。「マシン語のプログラムを楽しんでください」というのではないわけですね。このBASICについては，あとでもう一度触れたいと思います。

また，16ビットボードキットですから，拡張スロットに「ポン!」と差し込んで使うものではありません。ボードそのものが

大きいから，というのもそうですが，拡張スロットを持っていない人にも使えるようにと配慮されているのでしょう。そのかわり，工具（プラスドライバー）や改造（とは言ってもハンダ付け，パターンカットなどはいっさいナシです）に必要なネジや金具まで一式揃っています。

このキットの価格は¥78,000で，もちろんマニュアル（180ページ）も付いてきます。

16ビットボードの取り付け

16ビットボードの取り付け方法について，かんたんに見ておきましょう

①MZ-2000のネジを外します（グラフィックRAMを取り付けたときの要領で)。
②キーボードを外し，まん中の支柱とスピーカーを外します。
③電源トランス（付属）を取り付けます。
④Z-80 CPU を手持ちのマイナスドラバーなど，先の細いもので外します。
⑤キーボードの下のすき間に16ビットボードを置き，付いているケーブルの先をZ-80のソケットに取り付けます。ボードもネジで固定します。
⑥スピーカーをブラウン管の隣り付近に，金具とともに固定します。
⑦キーボードとフタ・ネジをえのように戻します。

結構やさしそうですね。私は30分もかかりませんでしたが，ドライバーをほとんど握ったことのない人でも，1時間もあれば十分でしょう。特にこれといって失敗しそうな部分はありませんでしたが，私はスピーカーの取り付けにやや時間をとられました。大部分は，A4版，12ページの図入り説明がなされているおかげで，まず大丈夫なはずです。

なお，漢字ROMも購入した人は，①の前にボード本体に取り付ければ良いことになっています。もちろん，あとで取り付けることもできます。

これらの作業に自信のない人は，ハードウェアの技術のしっかりしたマイコンショップで購入し，ついでに取り付けてもらうと良いでしょう。

16ビットBASICの詳細

先に触れたとおり，「16ビッドボードがありますからマシン語で楽しんでください」では，ほとんど利用できる状態ではありません。ハードを十分に活用できる BASIC が付いてきてこそ，初めて意味のある使い方ができるというものでしょう

本キットに付属のBASIC「1Z-012」は，MZ-2000に付属のMZ-1Z 001と比べて，次のような特徴があります。

①カラーグラフィックを扱えます。
②CIRCLE, PAINT, BOXなどのグラフィック命令が使えます。
③漢字をサポートするいくつかの命令や関数が追加されています（もちろん漢字ROMカードがないと動きません)。
④フリーエリアが，プログラムエリアとして64906バイト，データ（変数）エリアとして38000バイト，合計102906バイトも使けるようになりました。
⑤④に伴い，配列変数の要素が1次元のときに限り，メモリ範囲内（今までは255）に拡大されました。
⑥MZ-1Z 001のプログラムテープは，そっくり読み書きができます。

まず，①と②とは，MZ-1Z 002，すなわち**COLOR BASICに共通**です。COLOR BASICを持っていない人は，もうこれで購入する必要がなくなりましたし，すでに持っている人も，これまでに作っておいたプログラムが役に立つわけです。

③はまったく新しい命令，KPATTERN 文・JTRANS関数・KCHR$関数のことです。KPATTERN文はPRINT 文の漢字版で，指定漢字コードのパターンをグラフィック画面に表示します。PATTERN文は，

PATTERN −16, A$

（ただしA$は16の整数倍の長さのストリングで，ビットデータを持つ）

としていたので，漢字のビットデータがわかる必要があったうえに，一般にLEN(A$）は32ですから，最大7個までの漢字しか1変数に持つことができません。

1Z-012のKPATTERN文では，漢字1個＝2バイト（従来の2文字分）で扱いますから，1変数に127個までの漢字が代入できます。そして，従来の「文字型変数」を使用しますから，RIGHT$, LEFT$, MID$, LENなども利用できます。漢字1文字＝英数・カナ文字の2文字ということは，初めの3個の漢字を取り出すのに

A$=LEFT$（A$, 6）

（6は3＊2の意味です）

とする必要があるのですが，これでも相当使いものになりそうです。

従来のCHR$に相当するのが，KCHR$です。キャラクタコードは0～255ですが，漢字コードは8481～20350までありますので，CHR$を流用することができません。そうかといって変換するプログラムを，

A$＝CHR$（A−INT(A/256)＊256）
＋CHR$（INT(A/256)）

のようにして作っていたのでは，スピードもプログラムも効率が悪くなってしまいます。KCHR$は，上のプログラムを，

A$＝KCHR$(A)

と置き換えられ，スマートなプログラムが書ける便利な関数です。これを KPATTERN文と組み合わせれば，

KPATTERN KCHR$($4977),KCHR$($3456), KCHR$($2121), KCHR$($3940)

とするだけで，かんたんに

風間　浩

のように表示できます。もちろん，

A$＝KCHR$（$4977）＋KCHR$（$3456）＋KCHR$（$2121）＋KCHR$（$3940)

としておき，あとで，

KPATTERN A$

とすることもできますし，

KPATTERN LEFT$(A$, 4)

で，

風間

とだけ表示させることもラクラクですね。

もうひとつのJTRANSとは，要するにこんな感じです。

A=ASC（〝カ″）：A=JTRANS(A)：KPATTERN KCHR$(A)

カ

（カは，16＊16ドットで出る）

すなわち，引数のキャラクタコードを，相当する漢字コードに変換するもので，大きさを揃えて表示させたい場合に便利です。

A$＝〝″：B$＝〝カザマ″：FOR A＝1 TO 6：A$＝A$＋KCHR$（JTRANS（ASC(MID$(B$, A)))）：NEXT：KPATTERN A$,KCHR$($3940)

とすると，

カザマ　浩

となるのです。

なお，KPATTERN文は自動改行しませんから，PATTERN文と同じく，POSITION で位置を設定する必要がある場合もあります。KCHR$ の引数は，やはりCHR$と同じく，10進数，16進数，数値変数，数

表1　BASIC 16 (1Z-012) で, MZ-1Z001から追加された命令等。

	命令	例	解説	サンプルプログラムの解説
	KPATTERN	KPATTERN KCHR$($3026)	漢字のプリントを行なう	愛という文字を表示する
	JTRANS	KPATTERN KCHR$(JTRANS(ASC("A")))	キャラクタコード→JISコード変換	Aという文字を表示する
	KCHR$	PRINT ASC(KCHR$(1000))	JISコード→2文字の文字列	232($E8)とプリントする
	TABSET	TABSET 10,20,30,40,50,60,70	TAB位置の設定	10, 20, …70桁にTABをセットする
	SIZE1	? SIZE1	プログラムエリアのフリーサイズ	プログラムの残り容量をプリントする
	SIZE2	? SIZE2	変数エリアのフリーサイズ	変数の残り容量をプリントする
	DIM	DIM A$(10000)	DIMの添字が256以上でも良い	
カラーBASIC共通	COLOR	COLOR@ 1,7,0	グラフィックのカラー設定	カラー入力ページ1, 出力全ページ, W0
	CCOLOR	CCOLOR@ 4,0	テキスト, 背景色設定	テキストを緑, 背景色を0にする
	GRAPH	GRAPH C7	CとFが3ページ同時可能	全ページのクリア
	CONSOLE	CONSOLE P	PとMで, 重ね合わせ表示指定	内臓ディスプレイにグラフィックを重ねない
	SET	SET [4,W1] 95,40	〔色, Wn〕で, その場限りの色指定, 色の重ね合わせ指定ができる（SET～PATTERN）	
	RESET	RESET [7,W0] 261,3		
	LINE	LINE [2] 96,49,49,96		
	BLINE	BLINE [7] 0,0,639,199		
	PATTERN	PATTERN [1] -8,STRING$(CHR$(255),16)		
	PAINT	PAINT [4,W0] 320,100,1	色のぬりつぶし	(320,100)の, まわり（青で囲まれる）内を緑色でぬりつぶす
	BOX	BOX 10,10,629,189	箱を描く	LINE10, 10, 629, 10, 629, 189, 10, 189, 10, 10と同じ
	CIRCLE	CIRCLE 320,100,100	円を描く	(320,100)に半径100の円を描く
モニタコマンド	R		BASICのホットスタートにジャンプする(J CR 100,03 CR)	
	B		ブレークポイントを設定する	
	C		ブレーク後のCONT(続行)を行なう	

値式が使えます。

話は戻りますが, ④のフリーエリアの拡大は, おそらくこの16ビットボードの最大の特長だと思います。プログラム64K, 変数38Kなんて, そうかんたんに使えるものではありません。みなさんでこれに挑戦してみませんか？　ただ,「プログラムは38Kでいいから変数は64Kくれ！」という希望はかなえられません。あくまでもプログラムが最大64K, 変数が最大38Kなのです。

最後の⑤は, もっとも大切なところです。「マシン語で楽しんでください」同様,「1Z-012用に, キーボードから打ち直してください」では, はっきり言って使いものになりません。ただ, ここでさすがMZ！と言えるサポートがしてあり,

MZ-1Z001, MZ-1Z002

で作られたプログラムのすべてと,

MZ-1Z003（倍精度BASIC）
MZ-2Z001（デイスク）
MZ-2Z002（ディスクカラー）
MZ-2Z003（デイスク倍精度）
MZ-1Z001（テープGPIB）
MZ-80B用のすべてのBASIC）

で作られたプログラムのすべて（ない命令は除く）を, LOAD, RUNさせることができます。もちろんマシン語が入っていてはダメですが, 今まで自分で作ってきたプログラムには, あまりマシン語は使っていないでしょうから, そっくり利用できることと思います。

8ビット機としてのMZ

さて, いろいろと16ビットボードの良い点を見てきましたが, ここでひとつの疑問がわいてきた人もいるかと思います。

確かに16ビッドボードは良さそうだ。だが, MZが16ビットになってしまったらBASICでないソフトは動かないし, ちょっとしたプログラムをマシン語で組もうにも, まだ8088は理解できていないから, 不便じゃないかな？

といったところではないでしょうか, 1Z-012が, MZ-1Z001とMZ-1Z002を含む内容だとしても, 倍精度BASICやディスクBASIC, FloppyDOSなどは発表されていないのです。

この疑問点に対しては, とても気持ち良く「心配無用」と言い切れます。というのは, プリンタを取り付けて動かなくなるソフトなどないのと同じで,「MZ-2000が16ビットに変身する」とは言っても, 実際はMZ-2000とワンボード16ビットコンピュータを接続するだけのことしかやっていないからです。要するに8088の載ったマイコンは, 単なる周辺装置でしかないわけで, 1Z012のテープを読み込んだときにだけ, Z-80が8088のスレーブCPUとなるからです。かんたんに言えば, 今までどおりMZ-1Z001をセットすればBASICが動き, MZ-2Z001を使えば, ディスクBASICを利用でるのです。

メモリマップ

1Z-012のテープはIPLからロード, すなわち電源ONと同時に読み込ませればOKです。それに要する時間は約4分13秒とやや長めですが, これはIPL的プログラム, Z-80用IOCS, 8088用BASICの3本を, 順次ロードさせているためです。これらの関係を図にすると図1のようになります。

図1で▨となっている部分は, マニュアルの後部に「付録」として付いてきます（マニュアルにはBASIC以外をすべて公開している。という感じの文章がありますが, 結構非公開部があるので注意してください)。おそらく, 読者の方々の大半は8088に接するのは初めてでしょうから, かなり活用できる資料だと思います。

IPLから3段落でロードすることは前に触れたとおりですが, 詳しく見ると,

①0000H～0451H
②0500H～2F40H
③8088側の　00400H～06165H

の順で読んでくるものです。8088側では, ①と②の部分は関係なく, Z-80からのコマンドを持ちます。ユーザーがMZ-1Z001やHuBASICを使うなら, この状態が電源OFF時まで続きます。③のロードが終了すると, Z-80は8088に対してデータ(BASIC16など）を送り, 8088にすべて転送し終わると, 今度は8088がマスター, Z-80がスレーブとなる, デュアル CPU システムに変わ

ります。その後は，Z-80と8088とは互いにインタラプト（割り込み）をかけ合いながら，データのやりとりを行ないます。

8088がZ-80に「やりなさい！」と命令するコマンドには，キーボード入力，ディスプレイ出力，プリンタ出力，カセット入出力（図1のワークエリア②は，これに使われると考えられる），グラフィック全般などがあり，これらはすべて，8088が直接行なうことはないのです。

キーボードからの1行入力も当然Z-80の仕事ですが，ファンクションキーが20個になったり，すべてのキーでオートリピートしたりの拡張があります。パスカルやフロッピーDOSを使ったことのある人なら，モニタ2Z-004Aを知っていると思いますが，その内容とそっくりです。したがって SHIFT でプリント（正確にはスクロールアップ）が遅くなったり……などの機能もまったく同じに使えます。

なお，8088がソートなどのプログラムにかかりっきりになり，しばらくの間，キー入力など（要するにZ-80分担の仕事すべて）が必要ないときに限り，Z-80と8088とを分離して，別々の動作をさせることができます。M- 1Z001などで作成したテープ（5000H以降に限る）をZ-80側にLOADで読み込み，10バイトぐらいのマシン語プログラム（マニュアルに明記）を実行するだけで，Z-80はそのプログラムの「実行開始アドレス」にジャンプします。さらにおもしろいことには，8088に用事ができたときには，やはり共通のマシン語サブルーチンをCALLする（BASICよりUSR文を使う）だけで，Z-80にインタラプトがかかって戻ってくるのです。用途はかなり限られるとは思いますが，ソートプログラム実行中に，Z-80側でゲーム（当然マシン語のみで，モニタを参照していてはならない。モニタは新しい1Z-012Bですので）をする，というのがもっともらしい使い方だと思います。

8088入門

8088を使って，直接マシン語プログラムをやってみたいという方もいらっしゃると思いますが，マニュアルにはCPUについての説明はほとんどありません。ということは，命令そのものについては専門書を別途購入しなくてはなりません（実際の使い方はアセンブルリストが十分に役立ちます）。しかし，8086／8088のやさしい解説書をあまり見かけませんので，ここで，さわり程度に説明したいと思います。Z-80を知っている方も多いでしょうから，特に相違点を述べたいと思います。

まず，Z-80のレジスタは図2左のとおり22個あります。8088（8086）は18個です。ただ，バイト数にすればZ-80は26バイト，8088は28バイトですので，ごくわずかですが増えています。

8088のAL, CX, DX, BXは，Z-80のA, BC, DE, HLに相当し，8ビットの演算用にも16ビットの演算用にも使えます。Z-80ではBCやDEは，LD A,(BC)のようなポインタとしても使いましたが，8088では，BX(HL相当）でしか行なわれません。そのかわり，LDIRに相当するMOVSB 命令（ストリング操作のひとつ）では，SI, DIレジスタが使われ，SIがHLに，DIがDEに相当します。SIとDIは単独でポインタとして使われることもあります。そしてZ-80では，LD HL,0 ADD HL, SPとしたあとでスタック操作をすることがありますが，8088では，BPレジスタが主にこの用途に使われます。DXレジスタはZ-80のDEレジスタに相当するほか，16ビット＊16ビットや32ビット／16ビットの乗除算時に，AXの上位ビット群として使われたり，IN A,(C)で使われるBCレジスタに相当する働きも持っています。

Z-80では，HLレジスタをポインタとして使う場合にのみ，＋－数10バイトの範囲をインデックスして指定できるともいうべきレジスタ，IXとIYがありましたが，いくつか使いづらい点もありました。それは，IX, IYはHLとほぼ同じ16ビット操作（ポインタ操作）ができるものの，オフセット

図2　レジスタ構成

Z-80のレジスタ		8088(8086)のレジスタ	解　説
A	F	(AH)　A　X　(AL)	アキュームレータ。ALはAに相当する
B	C	(CH)　C　X　(CL)	カウントレジスタ。補助アキュームレータとしても使う。主にカウント用に使う。CHはB, CLはC, CXはBCに相当する。
D	E	(DH)　D　X　(DL)	データレジスタ。補助アキュームレータとしても使う。AXの上の16ビットとして乗除算に使ったり，16ビットI/Oにも使う。DHはD, DLはE, DXはDEに相当する。
H	L	(BH)　B　X　(BL)	ベースレジスタ。補助アキュームレータとしても使う。主にDSと加算して，メモリの一部を参照することもできる。BHはH, BLはL, BXはHLに相当する。他のセグメントレジスタと加算することもできる。
SP		SP	スタックポインタ。スタックエリアは，SSと加算して決まる。他のセグメントレジスタと加算することはできない。SPと同じに使う。
PC		IP	インストラクションポインタ。CSと加算して，命令を取り込む。他のセグメントレジスタと加算することはできない。PCと同意。
		FG (PSW)	ステータスフラグ(プロセッサステータスワード)。FGは，Fに相当するが，ビット単位では互換性がない。フラグは9個ある。
A'	F'	BP	ベースポインタ。主にSSと加算してスタック領域をアクセスするが，別のセグメントレジスタと加算することもできる。BPは，ポインタとして使う。SPに相当する。
B'	C'	DI	ディスティネーションインデックスレジスタ。主にDSと加算して，メモリをアクセスできる。ESとのみ加算してストリング操作ができる(ポインタとして使うDEに相当する)。BXやBPのインデックスにもなる。
D'	E'	SI	ソースインデックスレジスタ。主にDSと加算して，メモリをアクセスできる。主にDSと加算してストリング操作ができる(ポインタとして使うHLに相当する)。BXやBPのインデックスにもなる。
H'	L'		
IX		DS　0	データセグメント。BX, BP, DI, SIと加算して，メモリをアクセスする。ストリング操作時は，SIと加算するのが普通(他のセグメントも使えるが)。
IY		SS　0	スタックセグメント。BX, BP, DI, SIと加算して，メモリ(スタック内)をアクセスする。SPと加算し，スタックとして使う。
R	I	CS　0	コード(インストラクションコード)セグメント。BX, BP, DI, DSと加算して，主プログラム中のデータをアクセスする。IPと加算して，命令を取り込む。
8ビット長	8ビット長	ES　0	エキストラセグメント。BX, BP, DI, SIと加算して，メモリをアクセスする。ストリング操作時に，DIと加算する。SIと加算することもできる。
		16ビット長	

互換性のあるレジスタ：相当するレジスタがある。(8080の全レジスタでもある)

互換性のないレジスタ：相当するレジスタがない。(Z-80独自のレジスタでもある)

図3　セグメントレジスタの使われ方

を使わなくてもバイト数(メモリ)を多く消費し，かといってHLレジスタとデータ交換をしようにも時間・メモリを多く使います。さらにオフセットは定数しか指定できないため，単なるレジスタとしてのみ使われていたこともありました（まったく使っていないBASICも)。要するに，HLレジスタに定数・変数(＝レジスタの内容)でオフセットをつけられれば良いのですが，8088では，HLに相当するBXとスタック操作用のBPで，それが可能となりました。これなら必要なところでのみ，メモリ・時間を使ってオフセットを使えます。しかもレジスタの加算はその場限りのものですから，たいへん便利です。変数でのオフセットには，SIとDIの各レジスタが使われます。SIやDIの片方と定数との和をオフセットすることもできます。

8088の残りのレジスタは，セグメントレジスタです。8088のアドレス空間は1Mバイトですが，レジスタはみな16ビットなので，そのレジスタに関しては同時に64Kまでしかアクセスできません。これではZ-80と変わりなくなってしまいます。せめて，その64Kについては1Mバイトのうち，どこでも指定できるように，というわけで，セグメントレジスタが存在するのです。そしてセグメントレジスタは，本来なら20ビットにする必要があるのですが，1Mバイト中の数バイトなんて，東京〜大阪間の数メートルというぐらい小さい値ですから，20ビット中の下位4ビットは常に0ということにして，上位16ビットのみ指定することにしています。要するにDSレジスタに0040Hを入れれば，00400Hを意味するわけですね（図3参照)。

8088の命令名は，Z-80のそれと違うものがたくさんあります。代表的なものには，LD→MOV(MOVEの意)，JP→JMP，JP Z,→JZ(他も同様)，SBC→SBB(Bはボローの意)，CP→CM，DJNZ→LOOP(レジスタはCXを使用）があり，さらに「メモリの内容」を意味する( )は，[ ]となっています。

全体的に見ると，汎用レジスタは減少し，メモリ空間は増えたことがわかります。命令数はたいして変わっていないものの，[BX＋10］のようなオフセット使用が大幅に許されたため，展開すると相当な数になります。当然その多くはほとんど使われず，またZ-80では1バイトで良かったものが，2バイト命令になったものが増えて，Z-80と共通の命令のみを使えば，オブジェクトコードが増加します（ちなみに最大の命令は6バイト命令です)。

もちろん，8088を十分に使いこなしたプログラムなら，メモリ効率もスピードも強力なものにできますが，ここで注意しなくてはなりません。これは，コーディングレベルではもちろんのこと，アルゴリズムに関しても，8ビット向きのアルゴリズムと16ビット向きのアルゴリズムがありますからかなり複雑です。ことにBASICインタプリタのような，スピードをかなり重要視するものでは，8088を生かしたものにしなければなりません。シャープBASICはZ-80を十分に生かしたBASICとして，他メーカー機にかなり差をつけているのですが，1Z-012はそれをそっくり移植してしまっただけのようで，なんとMZ-1Z-001より20数パーセントもスピードダウンしています（汎用レジスタの減少が主原因だと思いますが)。

メモリ効率は，BASICの容量からみるとZ-80とほぼ同じですが，デュアルCPU構成にしたため，モニタ部ではやはり効率ダウンしています。もちろん，他メーカーの8088をCPUとしているほとんどの16ビット機も，Z-80の機種より遅いのですから，シャープだけを責める気はないのですが。もしかすると，BASICは8ビットCPUに適しているのかもしれません。

総合評価

本ボードは，78,000円と比較的安い値段で16ビットが体験でき，しかもBASICのユーザーエリアが倍以上になる（スピードダウンは20数パーセントで済む）とあって，久びさの大ヒット周辺機器になりそうです。ユーザーエリアを倍にするソフトがあったらこの値段でも……という方も中にはいらっしゃるはずですから，かなり安価だと考えて良いかもしれません。

ただ，ここで，いくつか注文をつけたいと思います。メーカーの方々にぜひとも実現してほしいと思うのですが，いかがなものでしょうか。

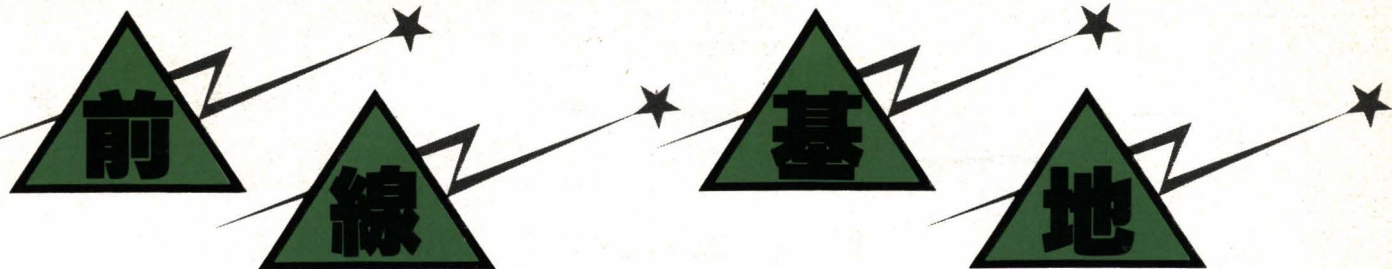
まず，今まで出してきた倍精度BASIC，システムプログラム，ディスクBASIC，フロッピーDOS，マシンランゲージなどを，16ビットカード用としても発売してほしいということ。また，フロッピーDOSは，Z-80モードで良いので，8088クロスアセンブラを販売してほしいものです。明らかにフロッピーDOS＋クロスアセンブラによる出力(つまり開発は終わっている)ですから，今すぐにでもできるのではないでしょうか。ディスクBASICも早急にお願いしたい。マスターディスケット内部に，フロッピーDOS用クロスアセンブラを入れておく，というのはどうでしょうか。これだけ広大なメモリには，ディスクはかなり必要です。

そのほか，＋128Kバイトのメモリ増設カードを出せば，HuBASIC-86（8086用のもので，128Kバイトもの大規模BASIC。当然8088でも使える）が載ると思います。漢字ROMを含み，漢字ROM用ソケットに差し込んで使えると思うのですが。これは，周辺ハードメーカーにも期待したいですね。

BASICについては，プログラムエリア38Kと，変数エリア64K（インタプリタ内の関数用定数は，起動時に変数エリア先頭に転送しておけば，セグメント管理も容易なはず）の組み合わせのほうがベターだと思います。プログラムを38K作れる人はいなくても，データを38K使いきるのはかんたんですからね。

細かなところでは，KCHR＄で1文字のみを変換するのではなく，JTRANS＄(A＄)でA＄の内容が拡大表示可能にすれば，ループを回す必要もなく，もっと便利だと思います。2次元配列でも255の最大要素数を拡大したり，Z-80のサブルーチンをかんたんに呼び出せるようなPOKE@，PEEK@，USR@文なども追加してもらえると，かなり高度な応用ができるでしょう。これらは，クリーンコンピュータならテープの交換のみで実現できるものですから，将来を楽しみにしたいと思います。

そして，マニュアルは，8088の命令をもう少し説明してほしかったですね。策としては，マシンランゲージにでも同梱してもらえるとよいのですが。

最後にもうひとつ。それは，MZ-80B用の16ビットボードも発売してほしいということ。ちょっとの時間の差でMZ-80Bを購入した人も多く，また，Bのユーザーは相当高度な活用をしている人が多いはずですから，せめて周辺装置では2000と差別されたくないのです。この16ビットボードキットが，単なるイメージアップ商品でないことを形づけるためにも，ぜひ実行してもらいたいものです。

★Elder Man氏のコメント

おもしろゲーム1等賞入賞の知らせを聞いて本当に驚きました。内心では上位入賞はねらってはいたのですが，ALL BASICプログラムですからどこまでいけるかはまったく予想できませんでした。

ゲームのアイデアはX1のBASICを勉強しているうちに考えついたもので，X1に標準装備されているPCG，PSG，さらにGRAPHIC機能を活用すれば，これまでのゲームにみられなかった効果が得られると思いました。また，BASICの実行速度の遅さの問題ですが，敵に照準を合わせる作業を画面上で実行すると，照準の移動のための時間が必要になるため，ゲームのスピード感が損われてしまいます。この前線基地ではその問題の解決のため，キーボードをフルに使用したモグラタタキ型の照準合わせを採用してみました。

これからも，もっともっとおもしろい，新しいゲーム作りに励みたいと思います。どうもありがとうございました。

★ゲームの方法

「1943年8月　フランス

ここは，連合軍の前線基地です。

ノルマンディー上陸ののち，連合軍は一勢攻撃の準備を整えています。

ドイツ軍は，猛反撃を開始しました。」

という状況設定です。

スクリーンには，緑の大地と遠くの山々，青い空と白い雲，手前には前線基地を守るための「土のう」が描かれています。

山の谷間から迫ってくるドイツ軍のタイガー戦車軍団やユンカース爆撃機をキャノン砲で撃破して基地を守るというゲームになっています。

ドイツ軍の攻撃は朝8時から始まり，深夜にまで及びます。7回にわたるドイツ軍の攻撃を撃破すると，夜が白々と明け，連合軍の勝利となります。

★キーの操作方法

①キャノン砲の発射キー

敵を攻撃するための武器としてキャノン砲が用意されています。画面のこちら側にあるため姿は見えませんが，キーで指示したところを攻撃します。発射キーとして，下のアルファベットキーの全部を使います。

左から右までのキーの位置が，スクリーン上の左から右までに対応していて，キーの配置が，そのままターゲットになっています。さらに数値キーの1～6で全体のポジションを上下させることができ，画面全体をねらうことができます。モグラタタキの要領でねらうと考ればわかりやすいと思います。

②爆破スイッチ

前線基地の「土のう」の向こうには，ドイツ軍の戦車が接近した際の最後の切り札として，爆破工作員がTNT火薬を7か所にしかけます。それらの爆破スイッチが「＋」キーで，7ヵ所のすべてが爆破し，近くにいる戦車を破壊するものです。TNTは破壊力は大きいのですが，場所を変更することができないので，プレイヤーはしかけられた場所をよく覚えておく必要があります。TNTは一度しか使用できません。明るい時間帯では，次の戦いの始まる前に再び爆破工作員がTNTをしかけますが，夜になるとしかけられません。夜戦に備えて，使用せずにおく作戦も必要になります。

③照明弾発射キー

昼間は役に立ちませんが，夜戦に入ると敵の姿がみえないので使用します。敵の位置を確認するために照明弾を発射するキーは「＝」です。

照明弾を使用した戦闘シーンがこの前線基地での闘いのクライマックスになります。

★ゲームの終了

①前線基地に築かれた「土のう」が2/3以上破壊されてしまったとき（敗戦）

②ドイツ軍の戦車に接近されてしまったとき（敗戦）

③戦いに勝ちぬいて敵をせん滅したとき（勝利）

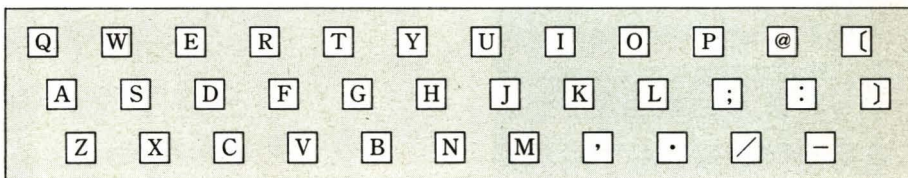
★必勝法

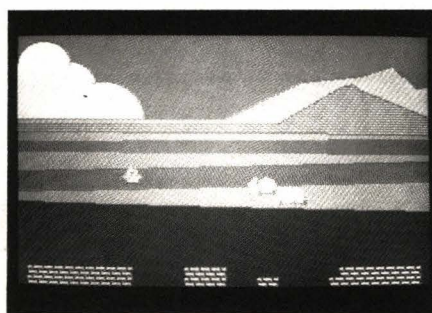
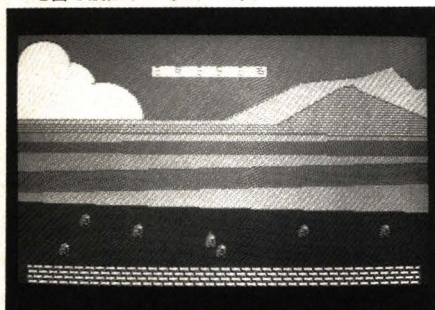
①正確な射撃のために練習を積むこと。夜戦に備え，ねらった位置に正確に当てられるようテクニックを身につける。

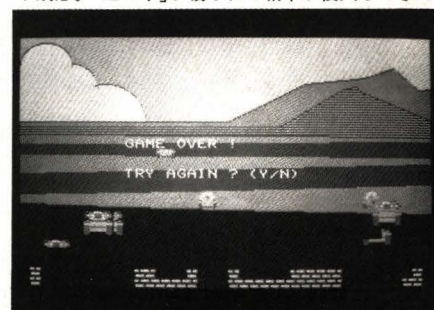
②「土のう」が破壊されてしまったところは，何回破壊されても同じなので，新たな部分を破壊されないように優先順位をつけて攻撃する。

★変数名と内容

変数名	内容
TX( )，TY( )	戦車の座標（移動後）
RX( )，RY( )	〃　（移動前）
C(0)～C(39)	土のうの被害
JX( )，JY( )	TNTの設置座標
X.Y.L	キャノン砲の攻撃座標
FLG	GAME OVER判定
F	敵の現われる初期座標
NM	敵の数（全体）
N	〃　（同時に現われる数）
T	命中範囲の設定(1or2)

Q	W	E	R	T	Y	U	I	O	P	@	[
A	S	D	F	G	H	J	K	L	;	:	]
Z	X	C	V	B	N	M	,	.	/	―	

プログラム内容の概略	
行　番～号	内　　容
6 ～	初期設定
9 ～	ゲームスタート
11 ～	初期デモ（爆破工作員など）
20 ～	キー入力ルーチン
25 ～	戦車行動ルーチン
55 ～	爆撃機行動ルーチン
78 ～	キー入力処理サブルーチン
132 ～	キャノン砲撃サブルーチン
145 ～	キャノン砲命中判定，処理サブルーチン
153 ～	戦車の攻撃サブルーチン
162 ～	土のうの破壊サブルーチン
174 ～	ＴＮＴ設置サブルーチン
195 ～	ＴＮＴ爆破サブルーチン
210 ～	照明弾発射サブルーチン
223 ～	ラウンドセットサブルーチン
247 ～	ミュージック（戦闘開始）サブルーチン
259 ～	ミュージック（勝利）サブルーチン
277 ～	GAME OVER（敗戦）ルーチン
293 ～	GAME OVER（勝利）ルーチン
309 ～	効果音（SOUND）サブルーチン
327 ～	タイトル表示サブルーチン
341 ～	キー操作説明表示サブルーチン
362 ～	背景描出、ＰＣＧ定義サブルーチン
378 ～	ＰＣＧ定義用DATA
487 ～	プレーヤーのレベル設定サブルーチン

※サブルーチンを多用しているように見えますが，プログラムのブロック構成を分りやすくするためです。

▲命中！　戦車を砲撃したが…

▼地雷を設置する爆破工作員
▼残念。「土のう」が破られて戦車が侵入してきた

```
1 REM ****************************************************************
2 REM
3 REM *  ZENSEN KICHI  COPYRIGHT 1983  by Elder Man  Vr.1.0  *
4 REM
5 REM ****************************************************************
6 WIDTH 40:CLEAR:CLICK OFF:INIT:PRW:DEFINT A-Z:WINDOW
7 DIM TX%(20),TY%(20),RX%(20),RY%(20),C%(39),JX%(7),JY%(7),JK%(4)
8 LOCATE 10,10:PRINT "WAIT A MINUTE !":MUSIC"V805B1":GOSUB 362
9 GOSUB 327:CLS:SCREEN 0,0,0:COLOR7,0:GOSUB 487
10 PALET:K=0:I=0:E=1:JFLG=0:CGEN0:CLS
11 GOSUB 223
12 FOR J=2 TO 37:C(J)=1:NEXT:CT=37
13 CGEN1:LOCATE1,23:PRINT STRING$(38,"K")
14 LOCATE1,24:PRINT STRING$(38,"K");
15 COLOR7:IF K>1 GOSUB 247
16 GOSUB 174:GOSUB 309:FOR J=0 TO 1000:D=0:NEXT
17 IF K>5 THEN SCREEN 1,1,0
18 CGEN0:CONSOLE0,22:CLS:CONSOLE:CGEN1
19 SCREEN 0,0,0:L=0:U=1:FOR I=1 TO N:TX(I)=RND*35+1:TY(I)=F:NEXT
20 S$=INKEY$(0):S=ASC(S$)-41:FRG=0
21 ON S GOTO 78,79,80,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97
22 S=S-20:ON S GOTO 98,99,100,101,102,103,104,105,106,107,108,109,110,111,112,11
3,114,115,116,117
23 S=S-20:ON S GOTO 118,119,120,121,122,123,124,125,126,127,128,129,130,131
24 CGEN 1:ON K GOTO 25,25,25,55,25,25
25 GOSUB 309:IF D>NM THEN 45
26 IF FLG=1 THEN 277
27 I=I+1:IF I>N THEN I=1
28 TX(I)=TX(I)+RND*2-1:TY(I)=TY(I)+1
29 IF TX(I)<2 THEN TX(I)=2
30 IF TX(I)>35 THEN TX(I)=35
31 IF TY(I)>15 THEN 35
32 LOCATE RX(I),RY(I):PRINT "**":LOCATE RX(I),RY(I)+1:PRINT "**"
33 U=-U:IF U>0 THEN TY(I)=TY(I)-E:LOCATE TX(I),TY(I):PRINT ";<":LOCATE TX(I),TY(
I)+1:PRINT "=>":GOSUB 153:GOTO 43
34 LOCATE TX(I),TY(I):PRINT "?@":LOCATE TX(I),TY(I)+1:PRINT "AB":GOTO 43
35 LOCATE RX(I),RY(I):PRINT "***":LOCATE RX(I),RY(I)+1:PRINT "***":LOCATE RX(I),
RY(I)+2:PRINT "***"
36 LOCATE TX(I),TY(I):PRINT"LMN":LOCATE TX(I),TY(I)+1:PRINT "OPQ":LOCATE TX(I),T
Y(I)+2:PRINT "RST"
37 FOR J=0 TO 200:NEXT
38 LOCATE TX(I),TY(I):PRINT"UVW":LOCATE TX(I),TY(I)+1:PRINT "XYZ":LOCATE TX(I),T
Y(I)+2:PRINT "[¥]"
39 GOSUB 160:GOSUB 162
40 LOCATE TX(I),TY(I):PRINT"LMN":LOCATE TX(I),TY(I)+1:PRINT "OPQ":LOCATE TX(I),T
Y(I)+2:PRINT "RST"
41 IF TY(I)=20 THEN 277 ELSE 43
42 LOCATE RX(I),RY(I):PRINT "***":LOCATE RX(I),RY(I)+1:PRINT "***":LOCATE RX(I),
RY(I)+2:PRINT "***"
43 RX(I)=TX(I):RY(I)=TY(I)
44 GOTO 20
45 I=I+1:IF I>N THEN I=1
46 TY(I)=TY(I)-1
47 LOCATE RX(I),RY(I):PRINT "***":LOCATE RX(I),RY(I)+1:PRINT "***":LOCATE RX(I),
RY(I)+2:PRINT "***"
48 IF TY(I)<8 THEN TY(I)=8:GOTO 51
49 U=-U:IF U>0 THEN LOCATE TX(I),TY(I):PRINT ";<":LOCATE TX(I),TY(I)+1:PRINT "=>
":GOSUB 153:GOTO 51
50 LOCATE TX(I),TY(I):PRINT "?@":LOCATE TX(I),TY(I)+1:PRINT "AB"
51 RX(I)=TX(I):RY(I)=TY(I)
52 A=0:FOR J=1 TO N:A=A+TY(J):NEXT
53 IF A<=8*N GOSUB 293:GOTO 11
54 GOTO 20
55 GOSUB 317:IF D>NM THEN 45
56 IF FLG=1 THEN 277
57 I=I+1:IF I>N THEN I=1
58 TX(I)=TX(I)+RND*2-1:TY(I)=TY(I)+1
59 IF TX(I)<2 THEN TX(I)=2
60 IF TX(I)>35 THEN TX(I)=35
61 IF TY(I)>8 THEN 64
62 LOCATE RX(I),RY(I):PRINT "**"
63 LOCATE TX(I),TY(I):PRINT "__":GOTO 67
64 IF TY(I)=12 THEN MY=TY(I):RX=RX(I):GOSUB 325:SY=0:GOTO 68
65 LOCATE RX(I),RY(I):PRINT "***":LOCATE RX(I),RY(I)+1:PRINT "***"
```

おもしろゲーム　1等賞入賞の「前線基地」(X1 HuBASIC)は，定価3200円でカセットサービスいたします。
発売は6月末の予定です。ご購入される方は最寄りのパソコンショップでお買い求めください。

```
66 LOCATE TX(I),TY(I):PRINT "yz{":LOCATE TX(I),TY(I)+1:PRINT "|}‾"
67 RX(I)=TX(I):RY(I)=TY(I):GOTO 20
68 TY(I)=TY(I)-1:IF TY(I)<0 THEN 76
69 GOSUB 325
70 LOCATE RX,RY(I):PRINT "ЖЖЖ":LOCATE RX,RY(I)+1:PRINT "ЖЖЖ"
71 LOCATE TX(I),TY(I):PRINT "▄▄▄":LOCATE TX(I),TY(I)+1:PRINT "███"
72 IF FGG=1 THEN 75
73 MY=MY+1:IF MY=23 GOSUB 162:FGG=1:SY=0:GOTO 75
74 LOCATE TX(I)+1,SY:PRINT "Ж":LOCATE TX(I)+1,MY:PRINT "π";
75 SY=MY:RY(I)=TY(I):RX=TX(I):GOTO 68
76 LOCATE RX,RY(I):PRINT "ЖЖЖ":LOCATE RX,RY(I)+1:PRINT "ЖЖЖ"
77 TX(I)=RND*35+1:TY(I)=0:FGG=0:FRG=0:GOTO 20
78 GOTO 24
79 GOSUB 195:GOTO 24
80 X=26:Y=2:GOSUB 132:GOTO 24
81 GOTO 24
82 X=29:Y=2:GOSUB 132:GOTO 24
83 X=32:Y=2:GOSUB 132:GOTO 24
84 GOTO 24
85 L=15:SOUND7,63:MUSIC"V1304C0R":GOTO 24
86 L=12:SOUND7,63:MUSIC"V1304E0R":GOTO 24
87 L=9:SOUND7,63:MUSIC"V1304G0R":GOTO 24
88 L=6:SOUND7,63:MUSIC"V1305C0R":GOTO 24
89 L=3:SOUND7,63:MUSIC"V1305E0R":GOTO 24
90 L=0:SOUND7,63:MUSIC"V1305G0R":GOTO 24
91 GOTO 24
92 GOTO 24
93 GOTO 24
94 X=34:Y=1:GOSUB 132:GOTO 24
95 X=31:Y=1:GOSUB 132:GOTO 24
96 GOTO 24
97 SCREEN0,0,0:GOSUB 210:GOTO 24
98 GOTO 24
99 GOTO 24
100 X=33:Y=0:GOSUB 132:GOTO 24
101 X=4:Y=1:GOSUB 132:GOTO 24
102 X=17:Y=2:GOSUB 132:GOTO 24
103 X=11:Y=2:GOSUB 132:GOTO 24
104 X=10:Y=1:GOSUB 132:GOTO 24
105 X=9:Y=0:GOSUB 132:GOTO 24
106 X=13:Y=1:GOSUB 132:GOTO 24
107 X=16:Y=1:GOSUB 132:GOTO 24
108 X=19:Y=1:GOSUB 132:GOTO 24
109 X=24:Y=0:GOSUB 132:GOTO 24
110 X=22:Y=1:GOSUB 132:GOTO 24
111 X=25:Y=1:GOSUB 132:GOTO 24
112 X=28:Y=1:GOSUB 132:GOTO 24
113 X=23:Y=2:GOSUB 132:GOTO 24
114 X=20:Y=2:GOSUB 132:GOTO 24
115 X=27:Y=0:GOSUB 132:GOTO 24
116 X=30:Y=0:GOSUB 132:GOTO 24
117 X=2:Y=0:GOSUB 132:GOTO 24
118 X=12:Y=0:GOSUB 132:GOTO 24
119 X=7:Y=1:GOSUB 132:GOTO 24
120 X=15:Y=0:GOSUB 132:GOTO 24
121 X=21:Y=0:GOSUB 132:GOTO 24
122 X=14:Y=2:GOSUB 132:GOTO 24
123 X=6:Y=0:GOSUB 132:GOTO 24
124 X=8:Y=2:GOSUB 132:GOTO 24
125 X=18:Y=0:GOSUB 132:GOTO 24
126 X=5:Y=2:GOSUB 132:GOTO 24
127 X=36:Y=0:GOSUB 132:GOTO 24
128 GOTO 24
129 X=37:Y=1:GOSUB 132:GOTO 24
130 GOTO 24
131 X=35:Y=2:GOSUB 132:GOTO 24
132 COLOR7:LOCATE X,Y+L:PRINT "/0":LOCATE X,Y+L+1:PRINT "12"
133 GOSUB 145:IF FRG=1 THEN SCREEN 0,0,0:LOCATE X,Y+L:PRINT "GH":LOCATE X,Y+L+1:
PRINT "IJ":FOR J=0 TO 100:NEXT:GOTO 135
134 GOSUB 143
135 LOCATE X,Y+L:PRINT "+,":LOCATE X,Y+L+1:PRINT "-."
136 FOR J=0 TO 100:NEXT
137 LOCATE X,Y+L:PRINT "34":LOCATE X,Y+L+1:PRINT "56"
138 FOR J=0 TO 100:NEXT
139 LOCATE X,Y+L:PRINT "78":LOCATE X,Y+L+1:PRINT "9:"
140 FOR J=0 TO 100:NEXT
141 LOCATE X,Y+L:PRINT "ЖЖЖ":LOCATE X,Y+L+1:PRINT "ЖЖЖ":LOCATE X,Y+L+2:PRINT "ЖЖ
Ж"
142 RETURN
143 SOUND 6,31-L/3:SOUND 7,55:SOUND 8,31:SOUND 11,185:SOUND 12,20:SOUND 13,9
144 FOR J=0 TO 200:NEXT:RETURN
```

```
145 FOR M=1 TO N
146 IF  ABS(X-TX(M)-G)<2 AND ABS(Y+L-TY(M))<T GOSUB 148:TX(M)=RND*38:TY(M)=F:FRG
=1:D=D+1
147 NEXT:RETURN
148 SOUND 3,31:SOUND 7,7:SOUND 8,31:SOUND 9,31:SOUND 10,31:SOUND 11,185:SOUND 12
,20:SOUND 13,9
149 FOR J=0 TO 100:NEXT
150 SOUND 3,31:SOUND 7,7:SOUND 8,31:SOUND 9,31:SOUND 10,31:SOUND 11,185:SOUND 12
,20:SOUND 13,9
151 LOCATETX(M),TY(M):PRINT"***":LOCATETX(M),TY(M):PRINT"***"
152 D=D+1:RETURN
153 IF RND>.3 THEN RETURN
154 IF TY(I)>15 THEN LOCATE TX(I),TY(I):PRINT"UVW":LOCATE TX(I),TY(I)+1:PRINT "X
YZ":LOCATE TX(I),TY(I)+2:PRINT "[¥]":GOTO 160
155 LOCATE TX(I),TY(I):PRINT "CD":LOCATE TX(I),TY(I)+1:PRINT "EF"
156 SOUND 6,20:SOUND 7,55:SOUND 8,31:SOUND 11,185:SOUND 12,20:SOUND 13,9
157 FOR J=0 TO 200:NEXT
158 LOCATE TX(I),TY(I):PRINT "?@":LOCATE TX(I),TY(I)+1:PRINT "AB"
159 GOSUB 162:RETURN
160 SOUND 6,20:SOUND 7,55:SOUND 8,31:SOUND 11,185:SOUND 12,20:SOUND 13,9
161 FOR J=0 TO 200:NEXT:RETURN
162 IF RND<.5 THEN B=1 ELSE B=0
163 SCREEN 0,0,0
164 COLOR7:LOCATE TX(I),21+B:PRINT "^_`":LOCATE TX(I),22+B:PRINT "abc":LOCATE TX
(I),23+B:PRINT "def";
165 SOUND 6,31:SOUND 7,7:SOUND 8,31:SOUND 9,31:SOUND 10,31:SOUND 11,185:SOUND 12
,20:SOUND 13,9
166 FOR J=0 TO 100:NEXT
167 LOCATE TX(I),21+B:PRINT "ghi":LOCATE TX(I),22+B:PRINT "jkl":LOCATE TX(I),23+
B:PRINT "mno";
168 C(TX(I))=0:C(TX(I)+1)=0:C(TX(I)-1)=0
169 LOCATE TX(I),21+B:PRINT "pqr":LOCATE TX(I),22+B:PRINT "stu":LOCATE TX(I),23+
B:PRINT "vwx";
170 CT=0:FOR J=2 TO 37:CT=CT+C(J):NEXT
171 IF CT<13 THEN FLG=1
172 LOCATE TX(I),21+B:PRINT "***":LOCATE TX(I),22+B:PRINT "***":LOCATE TX(I),23+
B:PRINT "***";
173 RETURN
174 IF K>5 THEN RETURN
175 IF JFLG=1 THEN RETURN ELSE JFLG=1
176 SOUND 7,63:FOR J=1 TO 7:JX(J)=RND*34+2:JY(J)=18+INT(RND*4):NEXT
177 CGEN 1:TEMPO 200:FOR J=1 TO 7:FOR JJ=0 TO JX(J) STEP 2
178 GET@ (JJ,JY(J)-1)-(JJ,JY(J)),JK%
179 LOCATE JJ,JY(J)-1:PRINT"|":LOCATE JJ,JY(J):PRINT"┤"
180 MUSIC "V806B0":PUT@ (JJ,JY(J)-1)-(JJ,JY(J)),JK%
181 GET@ (JJ+1,JY(J)-1)-(JJ+1,JY(J)),JK%
182 LOCATE JJ+1,JY(J)-1:PRINT"┴":LOCATE JJ+1,JY(J):PRINT"├"
183 MUSIC "V806B0":PUT@ (JJ+1,JY(J)-1)-(JJ+1,JY(J)),JK%
184 NEXT:LOCATE JX(J),JY(J)-1:PRINT"┬":LOCATE JX(J),JY(J):PRINT"┼"
185 MUSIC "V1204G2":NEXT
186 FOR J=1 TO 7:LOCATE JX(J),JY(J)-1:PRINT"*":LOCATE JX(J),JY(J):PRINT"*"
187 FOR JJ=JX(J) TO 38 STEP 2
188 GET@ (JJ,JY(J)-1)-(JJ,JY(J)),JK%
189 LOCATE JJ,JY(J)-1:PRINT"|":LOCATE JJ,JY(J):PRINT"┤"
190 MUSIC "V806B0":PUT@ (JJ,JY(J)-1)-(JJ,JY(J)),JK%
191 GET@ (JJ+1,JY(J)-1)-(JJ+1,JY(J)),JK%
192 LOCATE JJ+1,JY(J)-1:PRINT"┴":LOCATE JJ+1,JY(J):PRINT"├"
193 MUSIC "V806B0":PUT@ (JJ+1,JY(J)-1)-(JJ+1,JY(J)),JK%
194 NEXT:NEXT:RETURN
195 IF JFLG=0 THEN RETURN ELSE JFLG=0
196 FOR J=1 TO 7:FOR M=1 TO I
197 IF  ABS(JX(J)-TX(M))<4 AND ABS(JY(J)-TY(M))<5 THEN TX(M)=RND*38:TY(M)=F:FRG=
1:D=D+1
198 NEXT:GOSUB 200
199 NEXT:RETURN
200 SCREEN 0,0,0
201 COLOR7:LOCATE JX(J),JY(J)-2:PRINT "^_`":LOCATE JX(J),JY(J)-1:PRINT "abc":LOC
ATE JX(J),JY(J):PRINT "def";
202 SOUND 6,31:SOUND 7,7:SOUND 8,31:SOUND 9,31:SOUND 10,31:SOUND 11,185:SOUND 12
,20:SOUND 13,9
203 FOR Q=0 TO 100:NEXT
204 LOCATE JX(J),JY(J)-2:PRINT "ghi":LOCATE JX(J),JY(J)-1:PRINT "jkl":LOCATE JX(
J),JY(J):PRINT "mno";
205 FOR Q=0 TO 100:NEXT
206 LOCATE JX(J),JY(J)-2:PRINT "pqr":LOCATE JX(J),JY(J)-1:PRINT "stu":LOCATE JX(
J),JY(J):PRINT "vwx";
207 FOR Q=0 TO 100:NEXT
208 LOCATE JX(J),JY(J)-2:PRINT "***":LOCATE JX(J),JY(J)-1:PRINT "***":LOCATE JX(
J),JY(J):PRINT "***";
209 RETURN
210 SOUND 6,15:SOUND 7,55:SOUND 8,31:SOUND 11,185:SOUND 12,20:SOUND 13,9
```

```
211 FOR J=0 TO 100:NEXT:COLOR 7
212 FOR J=21 TO 1 STEP -1:LOCATE 25,J:PRINT "| ":LOCATE 25,J+1:PRINT "ж"
213 SOUND 0,15:SOUND 1,1:SOUND 6,27:SOUND 7,62:SOUND 8,24:SOUND 9,0:SOUND 10,0
214 SOUND 11,J+9:SOUND 12,0:SOUND 13,14NEXT
215 FOR J=0 TO 6:LOCATE 25,1:PRINT "| "
216 SOUND 6,15:SOUND 7,32:SOUND 8,10
217 FOR JJ=0 TO 200:NEXT
218 LOCATE 25,1:PRINT "| "
219 SOUND 6,15:SOUND 7,32:SOUND 8,10
220 FOR JJ=0 TO 200:NEXT
221 NEXT:LOCATE 25,1:PRINT "ж"
222 RETURN
223 K=K+1:D=0:FLG=0
224 ON K GOTO 226,228,230,232,234,238,242,259
225 RETURN
226 T=2:F=8:G=0:N=5:NM=40-LB:S$="8 月 8 日 8 時"
227 GOSUB 246:RETURN
228 PALET 1,5:N=7:NM=50-LB:E=0:S$="8 月 8 日 12 時"
229 GOSUB 246:RETURN
230 PALET:E=0:N=4:NM=40-LB:S$="8 月 8 日 14 時"
231 GOSUB 246:RETURN
232 F=0:N=3:NM=60-LB:T=1:G=0:S$="8 月 8 日 16 時"
233 GOSUB 246:RETURN
234 PALET 1,2:PALET 2,0:PALET 3,0:PALET 4,2:PALET 5,2:PALET 6,3:PALET 7,3
235 E=0:COLOR 3:F=8:N=6:NM=60-LB:G=0:T=2
236 S$="8 月 8 日 18 時"
237 GOSUB 246:RETURN
238 PALET 1,0:PALET 2,0:PALET 3,0:PALET 4,2:PALET 5,0:PALET6,2:PALET 7,0
239 COLOR 3:SCREEN 1,1,0:E=1:F=8:N=6:NM=40-LB
240 S$="8 月 8 日 20 時"
241 GOSUB 246:SCREEN1,0,0:RETURN
242 PALET 1,0:PALET 2,0:PALET 3,0:PALET 4,2:PALET 5,0:PALET6,2:PALET 7,0
243 COLOR 3:SCREEN 1,1,0:E=0:F=8:N=5:NM=40-LB
244 S$="8 月 8 日 24 時"
245 GOSUB 246:SCREEN1,0,0:RETURN
246 CREV1:COLOR 6:LOCATE 13,3:PRINT S$:CREV:COLOR7:RETURN
247 SOUND 7,63:SOUND 8,0:SOUND 9,0:SOUND10,0:TEMPO 220
248 A1$="C2R0C1RF2R0A1R+C2R0+C1RA2R0F1R"
249 A2$="A2R0A1RF2R0A1R+C5R"
250 A3$="C2R0C1RC2R0C1RF5R"
251 A4$="C2R0C1RC2R0C1R-A7"
252 MUSIC "V1005"+A1$
253 MUSIC "V1005"+A2$+":V904"+A1$
254 MUSIC "V1005"+A1$+":V904"+A2$+":V1003"+A1$
255 MUSIC "V1005"+A3$+":V904"+A1$+":V1003"+A2$
256 MUSIC "V1005"+A1$+":V904"+A3$+":V1003"+A1$
257 MUSIC "V1005"+A4$+":V904G2R0G1RG2R0G1RF7:V1003C2R0C1RC2R0C1RE7"
258 RETURN
259 SCREEN 0,0,0:PALET 4,3:PALET 6,3:TEMPO 100
260 S2$="C1RCRDRDR":S3$="G4E0RC0RCRERCR":S1$="-G0R-GR"
261 S4$="-A3F5D0R-BR":S5$="C6-G0R-GR":S6$="E4F0G1RFR":S7$="E1RERDRDR"
262 S8$="C6G0RFR":S9$="E4F0RG4F0R":S10$="E1RERERF0RER":S11$="D1RDRDRG0RFR"
263 MUSIC"V1006"+S1$+S2$+":V1104R1R6R1"+S1$:PALET 4,1:PALET 6,1
264 MUSIC"V1006"+S3$+":V1104"+S2$+":V1203"+":R6"+S1$:PALET 5,1
265 MUSIC"V1006"+S4$+":V1104"+S6$+":V1203"+S2$:PALET 1,1:PALET 7,5
266 MUSIC"V1006"+S5$+":V1104"+S7$+":V1203"+S3$:PALET 3,3:PALET 5,5
267 MUSIC"V1006"+S2$+":V1104"+S8$+":V1203"+S4$:PALET 4,4:PALET 6,4
268 MUSIC"V1006"+S9$+":V1104"+S10$+":V1203"+S5$:PALET
269 MUSIC"V1006"+S7$+":V1104"+S11$+":V1203"+S2$
270 MUSIC"V1006"+S5$+":V1104"+S10$+":V1203"+S3$
271 TEMPO 90:MUSIC"V1006"+S2$+":V1104D1RDRDRGR:V1203"+S4$
272 TEMPO 80:MUSIC"V1006"+S3$+":V1104E1RCRDRDR:V1203"+S5$
273 TEMPO 60:MUSIC"V1006-A3FE4R0G1R0+C7:V1104C3EG4R0+C1R0G7:V1203B3DF4R0E1R0E7"
274 CGEN 0:CREV1:COLOR6:LOCATE 5,5:PRINT "ｵﾒﾃﾞﾄｳ ! ﾄﾞｲﾂ ｸﾞﾝ ﾊ ｾﾞﾝﾒﾂ ﾃﾞｽ":BEEP
275 MUSIC"V1403C5EGCC:V1405+C5GECC"
276 COLOR7:CREV0:FLG=1:RETURN
277 ON K GOSUB 286,286,286,287,286,286,286
278 CGEN0:COLOR 7:LOCATE 10,10:PRINT "GAME OVER !":MUSIC "V805B1"
279 LOCATE 10,13:PRINT "TRY AGAIN ? (Y/N)"
280 S$=INKEY$:IF S$<>"" THEN 280
281 S$=INKEY$:IF S$="" THEN 281
282 IF S$="Y" OR S$="y" THEN 9
283 IF S$="N" OR S$="n" THEN LOCATE 10,16:PRINT "BYE BYE !!":MUSIC"05B1":GOTO 28
5
284 GOTO 281
285 SOUND 7,63:GOSUB 290:COLOR 7:END
286 SCREEN 0,0,0:FOR I=1 TO N:PALET4,2:GOSUB 154:PALET 4,4:GOSUB 162:NEXT
287 SOUND 7,62:SOUND 8,15:TEMPO 300:PALET 4,2
288 MUSIC "V1003+C1BABAGAGFGFEFEDEDCDC-BC-B-A-B-A-G7"
289 RETURN
290 SOUND 7,62:SOUND 8,15:TEMPO 300:PALET 4,2
```

```
291 MUSIC "V1003C1DEDEFEFGFGAGABAB+C7"
292 RETURN
293 CGEN0:CONSOLE0,22:CLS:CONSOLE
294 COLOR 1:LOCATE 10,10:PRINT "YOU WIN !!":COLOR7
295 SOUND 7,62:SOUND 8,15:TEMPO 300
296 MUSIC "V1004C2EGECEGEC7"
297 IF K<6 THEN CGEN1:GOSUB 300
298 FOR J=0 TO 4000:NEXT
299 CGEN0:CONSOLE0,22:CLS:CONSOLE:RETURN
300 CGEN1:FOR J=1 TO 200:PT=INT(RND*4+1)
301 IF J>100 THEN PT=4
302 ON PT GOSUB 305,306,307,308
303 PP=INT(RND*38)+1:LOCATE PP,23:PRINT P1$:LOCATE PP,24:PRINT P2$;
304 MUSIC "V1005G0":NEXT:RETURN
305 P1$="▌":P2$="■":RETURN
306 P1$="▌":P2$="/":RETURN
307 P1$="▌":P2$="-":RETURN
308 P1$="K":P2$="K":RETURN
309 ON K GOTO 311,313,315,317,319,321,323
310 RETURN
311 SOUND 2,160:SOUND 3,15:SOUND 6,31:SOUND 7,45:IF TY(I)>17 THEN SOUND 9,15 ELS
E SOUND 9,8
312 RETURN
313 SOUND 2,160:SOUND 3,15:SOUND 6,31:SOUND 7,45:IF TY(I)>17 THEN SOUND 9,15 ELS
E SOUND 9,9
314 RETURN
315 SOUND 2,160:SOUND 3,15:SOUND 6,31:SOUND 7,45:IF TY(I)>17 THEN SOUND 9,15 ELS
E SOUND 9,10
316 RETURN
317 SOUND 0,15:SOUND 1,2:SOUND 6,CT:SOUND 7,62:SOUND 8,24:SOUND 9,0:SOUND 10,0
318 SOUND 11,30:SOUND 12,0:SOUND 13,14:RETURN
319 SOUND 2,160:SOUND 3,15:SOUND 6,31:SOUND 7,45:IF TY(I)>17 THEN SOUND 9,15 ELS
E SOUND 9,10
320 COLOR 3:RETURN
321 SOUND 2,160:SOUND 3,15:SOUND 6,31:SOUND 7,45:IF TY(I)>17 THEN SOUND 9,15 ELS
E SOUND 9,10
322 COLOR 3:SCREEN 1,0,0:RETURN
323 SOUND 2,160:SOUND 3,15:SOUND 6,31:SOUND 7,45:IF TY(I)>17 THEN SOUND 9,15 ELS
E SOUND 9,10
324 COLOR 3:SCREEN 1,0,0:RETURN
325 SOUND 0,15:SOUND 1,1:SOUND 6,27:SOUND 7,62:SOUND 8,24:SOUND 9,0:SOUND 10,0
326 SOUND 11,33-TY(I):SOUND 12,0:SOUND 13,14:RETURN
327 COLOR 4,1:SCREEN 1,1,0:CLS:CSIZE 3
328 LOCATE 6,0:PRINT#0 "ZENSEN KICHI":COLOR 7
329 CSIZE0:LOCATE 0,4:PRINT "1943 年 8 月  ﾌﾗﾝｽ"
330 PRINT:PRINT:PRINT"ｺｺﾊ ﾚﾝｺﾞｳ ｸﾞﾝ ﾉ ｾﾞﾝｾﾝ ｷﾁ ﾃﾞｽ"
331 PRINT:PRINT:PRINT"ﾉﾙﾏﾝﾃﾞｨｰ ｼﾞｮｳﾘｸ.ﾉ ｱﾄ ﾚﾝｺﾞｳ ｸﾞﾝ ﾊ"
332 PRINT:PRINT"ｲｯｾｲ ｺｳｹﾞｷ ﾉ ｼﾞｭﾝﾋﾞ ｦ ﾄﾄﾉｴﾃ ｲﾏｽ"
333 PRINT:PRINT:PRINT"ﾄﾞｲﾂｸﾞﾝ ﾊ ﾓｳﾊﾝｹﾞｷ ｦ ｶｲｼ ｼﾏｼﾀ"
334 PRINT:PRINT:PRINT"ｱﾅﾀ ﾊ ｺﾉ ｾﾞﾝｾﾝ ｷﾁ ｦ ﾏﾓﾘﾇｲﾃ ｸﾀﾞｻｲ"
335 COLOR2:PRINT:PRINT:PRINT:PRINT"     ｾﾂﾒｲ ｶﾞ ﾋﾂﾖｳ ﾃﾞｽｶ ? (Y/N)"
336 GOSUB 247:COLOR7
337 I=1:TX(1)=35:TY(1)=16:CGEN1:GOSUB 154:GOSUB 164:CGEN 0
338 S$=INKEY$:IF S$="" THEN 338
339 IF S$="Y" OR S$="y" THEN 341
340 IF S$="N" OR S$="n" THEN RETURN ELSE 338
341 CLS:SOUND 7,63:MUSIC "V1005B1":COLOR4,1
342 PRINT "  *** ZENSEN KICHI by Elder Man ***"
343 COLOR6:MUSIC "V1305B1"
344 PRINT:PRINT"ｷｬﾉﾝ ﾎｳ ﾉ ﾊｯｼｬ ｷｰ ( ﾀｰｹﾞｯﾄ ﾆﾓ ﾅｯﾃｲﾏｽ )"
345 PRINT:PRINT" Q  W  E  R  T  Y  U  I  O  P  @  ["
346 PRINT"   A  S  D  F  G  H  J  K  L  ;  :  ]"
347 PRINT"    Z  X  C  V  B  N  M ,  .  /  _"
348 PRINT:PRINT"ﾀｰｹﾞｯﾄ ﾉ ｲﾄﾞｳ ｷｰ"
349 PRINT:PRINT"  1,2,3,4,5,6   ( LOW - HIGH )"
350 PRINT:PRINT"ﾊﾞｸﾊ ｽｲｯﾁ ( ﾊﾞｸﾊ ｺｳｻｸｲﾝ ﾉ ｼｶｹﾀﾓﾉ )"
351 PRINT"      +   ( 7 ｺ ﾄﾞｳｼﾞ ﾆ ﾊﾞｸﾊ ｼﾏｽ )"
352 PRINT:PRINT"ｼｮｳﾒｲ ﾀﾞﾝ ﾊｯｼｬ ｷｰ"
353 PRINT"      =   ( ﾖﾙ ﾆﾅｯﾀﾗ ﾂｶｲﾏｽ )"
354 COLOR7:CGEN1:PRINT:PRINT"KKKKK";:CGEN0
355 COLOR6:PRINT" ｶﾞ 2/3 ｲｼﾞｮｳ ﾊｶｲ ﾏﾀﾊ ｾﾝｼｬ ﾆ ｾｯｷﾝ"
356 COLOR7:CGEN1:PRINT"KKKKK";:CGEN0
357 COLOR6:PRINT " ｻﾚﾙﾄ GAME OVER ﾃﾞｽ"
358 COLOR2:PRINT:PRINT"CAPS LOCK ｼﾃｸﾀﾞｻｲ !"
359 COLOR7:LOCATE 5,24:PRINT "STAND BY OK ?  PUSH ANY KEY !";
360 S$=INKEY$:IF S$="" THEN 360
361 RETURN
362 FOR I=42 TO 150:A$="":FOR J=1 TO 24:READ A:A$=A$+CHR$(A)
363 NEXT:DEF CHR$(I)=A$:NEXT
364 SCREEN 0,0,0:PALET
365 LINE (0,132)-(319,140),PSET,4:PAINT (0,133),HEXCHR$("0000AA000055"),4
```

```
366 LINE (0,124)-(319,116),PSET,4:PAINT (0,125),HEXCHR$("00FF5500FFAA"),4
367 LINE (0,108)-(319,100),PSET,4:PAINT (0,109),HEXCHR$("00AA550055AA"),4
368 LINE (0,92)-(319,94),PSET,4:PAINT (0,93),HEXCHR$("00AAFF0055FF"),4
369 LINE (0,84)-(319,82),PSET,4:PAINT (0,85),4,4
370 LINE (0,76)-(319,78),PSET,4:PAINT (0,77),HEXCHR$("00FFAA00FF55"),4
371 COLOR4:LINE (0,70)-(200,70)-(250,40)-(260,38)-(264,42)-(319,64)
372 COLOR7:PAINT (0,71),HEXCHR$("00AA55AA55AA"),4
373 COLOR4:LINE (0,66)-(160,66)-(220,36)-(260,32)-(270,30)-(290,23)-(310,40)-(31
9,39)
374 COLOR7:PAINT (0,67),HEXCHR$("AAAA555555AA"),4
375 CIRCLE (20,20),20,4,.8,-10,180:CIRCLE (40,40),20,4,1,0,90:CIRCLE (66,66),30,
4,1,0,110
376 PAINT (20,20),7,4:PAINT (100,0),1,4
377 COLOR7,0:RETURN
378 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
379  DATA 0,0,32,2,0,34,0,1,0,16,41,11,116,46,91,47,0,16,9,9,48,12,82,13
380  DATA 0,0,0,0,0,72,160,64,0,0,72,52,144,236,248,244,0,0,64,16,144,164,24,160
381  DATA 0,6,18,0,4,16,0,0,55,39,95,27,38,21,12,1,51,2,77,18,2,5,8,0
382  DATA 144,96,144,32,0,0,0,0,246,248,242,244,76,82,160,0,214,216,98,144,76,64
,160,0
383  DATA 0,0,0,0,0,11,0,5,0,0,0,9,16,7,18,46,0,0,0,9,16,13,18,47
384  DATA 0,0,0,0,0,72,224,64,0,0,64,0,144,172,248,224,0,0,64,0,144,228,88,32
385  DATA 2,6,24,2,4,0,0,0,17,6,23,16,6,4,0,0,18,2,15,18,2,4,0,0
386  DATA 144,96,144,48,0,0,0,0,84,248,242,160,72,32,0,0,212,216,98,144,72,32,0,
0
387  DATA 0,0,32,2,0,34,0,0,0,16,33,10,84,34,65,42,0,16,8,0,0,4,64,5
388  DATA 0,0,0,0,0,72,32,0,0,0,72,36,144,108,40,84,0,0,64,0,0,4,128,96
389  DATA 0,0,18,0,4,0,0,0,22,33,86,1,38,0,8,1,0,2,64,16,0,5,0,0
390  DATA 0,32,128,0,0,0,0,0,130,100,146,68,128,18,136,0,0,144,32,0,8,0,128,0
391  DATA 0,0,0,0,0,0,0,0,16,40,2,8,34,64,5,0,0,0,0,0,0,0
392  DATA 0,0,0,0,0,0,0,0,64,16,0,72,164,64,0,0,0,0,0,0,0
393  DATA 0,0,0,0,0,0,0,0,4,66,16,4,9,0,0,0,0,0,0,0,0
394  DATA 0,0,0,0,0,0,0,128,16,160,0,8,0,128,0,0,0,0,0,0,0,0
395  DATA 0,0,3,7,15,15,15,15,0,0,3,0,0,0,32,0,0,3,7,15,12,14,47
396  DATA 0,0,0,224,240,240,240,240,0,0,0,0,0,0,4,0,0,0,224,240,48,112,244
397  DATA 0,4,32,80,32,95,32,0,127,123,95,47,95,32,80,0,127,123,15,15,15,0,0,0
398  DATA 0,32,4,10,4,250,4,0,254,222,250,244,250,4,10,0,254,222,240,240,240,0,0
,0
399  DATA 0,3,7,15,15,15,47,7,0,0,3,0,0,48,80,120,0,0,3,15,15,62,92,127
400  DATA 0,0,224,240,240,240,244,224,0,0,0,0,12,10,30,0,0,96,240,240,124,58,2
54
401  DATA 112,0,4,32,80,47,80,0,15,127,123,95,47,80,32,0,15,127,123,15,15,0,0,0
402  DATA 14,0,32,4,10,244,10,0,240,254,222,250,244,10,4,0,240,254,222,240,240,0
,0,0
403  DATA 0,0,3,21,34,0,35,21,0,40,95,190,127,255,126,190,0,0,3,29,54,41,54,29
404  DATA 0,0,0,168,68,0,196,168,0,20,58,125,254,255,126,125,0,0,0,184,108,148,1
08,184
405  DATA 0,4,32,80,32,95,32,0,127,123,95,47,95,32,80,0,35,115,15,15,15,0,0,0
406  DATA 0,32,4,10,4,250,4,0,254,222,250,244,250,4,10,0,196,206,240,240,240,0,0
,0
407  DATA 0,4,34,86,11,51,36,17,4,22,171,95,127,63,127,191,4,22,139,93,59,29,118
,157
408  DATA 0,0,144,192,104,88,160,73,128,32,216,244,250,252,250,253,128,32,144,20
8,250,180,26,169
409  DATA 72,30,50,36,157,26,32,0,127,191,127,63,191,95,46,37,123,154,109,54,155
,79,42,36
410  DATA 152,101,156,40,176,164,0,16,254,253,254,252,253,246,172,80,222,221,110
,152,253,228,168,80
411  DATA 0,231,231,0,0,126,126,0,231,24,24,231,126,129,129,126,0,231,231,0,0,12
6,126,0
412  DATA 0,0,0,1,3,3,3,3,0,0,0,0,0,0,0,0,0,1,3,3,3,3
413  DATA 0,0,124,255,129,255,153,129,0,0,84,0,0,0,0,0,0,84,255,255,129,231,25
5
414  DATA 0,0,0,128,192,192,192,192,0,0,0,0,0,0,0,0,0,128,192,192,192,192
415  DATA 3,1,0,0,0,84,40,84,0,24,3,127,127,43,84,43,3,24,127,127,127,3,3,3
416  DATA 255,255,0,0,129,0,0,0,0,255,255,126,255,0,255,255,0,255,255,126,255,
255,255
417  DATA 192,128,0,0,0,42,20,42,0,24,192,254,254,212,42,212,192,24,254,254,254,
192,192,192
418  DATA 40,84,43,84,40,0,0,0,87,43,84,40,84,0,0,0,3,3,0,0,0,0,0
419  DATA 0,0,255,0,0,0,0,255,255,0,0,0,0,0,255,255,0,0,0,0,0
420  DATA 20,42,212,42,20,0,0,0,234,212,42,20,42,0,0,0,192,192,0,0,0,0,0,0
421  DATA 0,0,0,0,0,9,0,0,0,4,82,23,47,95,47,23,0,0,18,5,42,29,42,5
422  DATA 0,0,124,127,0,255,24,129,0,0,85,128,129,129,129,0,0,0,84,255,255,129,2
31,255
423  DATA 0,0,0,0,0,144,0,0,0,32,74,232,244,250,244,232,0,0,72,160,84,184,84,160
424  DATA 3,1,0,0,0,84,40,84,0,24,3,127,127,43,84,43,3,24,127,127,127,3,3,3
425  DATA 126,255,0,0,129,0,0,0,129,0,255,255,126,255,0,255,126,0,255,255,126,25
5,255,255
426  DATA 192,128,0,0,0,42,20,42,0,24,192,254,254,212,42,212,192,24,254,254,254,
192,192,192
```

```
427  DATA 40,84,43,84,40,0,0,0,87,43,84,40,84,0,0,0,3,3,0,0,0,0,0,0
428  DATA 0,0,255,0,0,0,0,0,255,255,0,0,0,0,0,255,255,0,0,0,0,0,0
429  DATA 20,42,212,42,20,0,0,0,234,212,42,20,42,0,0,0,192,192,0,0,0,0,0,0
430  DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1
431  DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,68,42,0,0,0,0,0,0,4,32
432  DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,64,0,0,0,0,0,0,0,64
433  DATA 0,0,0,0,0,0,0,0,5,14,3,10,31,27,46,15,4,10,1,11,18,7,43,4
434  DATA 0,0,0,0,0,0,0,0,246,255,245,223,247,223,255,235,214,117,219,107,76,58,
163,212
435  DATA 0,0,0,0,0,0,0,0,128,160,192,160,240,80,248,112,128,32,192,32,192,208,4
8,144
436  DATA 0,0,0,0,0,0,0,0,23,15,63,23,15,15,3,0,22,3,41,22,11,5,2,0
437  DATA 0,0,0,0,16,8,0,0,191,253,127,255,253,127,255,254,99,154,197,40,19,201,
166,74
438  DATA 0,0,0,0,0,0,0,0,104,240,224,208,200,160,160,0,224,80,32,192,8,128,32,0
439  DATA 0,0,0,0,0,0,0,0,0,0,2,8,2,86,1,0,0,0,0,0,0,0,1
440  DATA 0,0,0,0,0,0,0,0,0,16,66,8,162,148,69,42,0,0,0,0,0,0,4,32
441  DATA 0,0,0,0,0,0,0,0,0,0,0,32,128,208,0,104,0,0,0,0,0,0,0,64
442  DATA 0,0,0,0,0,0,0,0,45,30,83,14,127,27,175,15,4,10,1,8,18,1,42,4
443  DATA 0,0,0,0,0,0,0,0,181,223,253,223,255,247,255,107,148,68,144,75,68,16,16
3,68
444  DATA 0,0,0,0,0,0,0,0,146,184,242,172,113,86,248,118,128,32,192,32,0,192,48,
16
445  DATA 0,0,0,0,0,0,0,0,85,15,63,23,14,11,3,0,16,3,40,6,8,1,2,0
446  DATA 0,0,0,0,17,40,16,0,255,253,127,255,254,255,255,254,35,24,69,40,16,104,
180,74
447  DATA 0,0,0,0,0,0,0,0,232,244,242,208,204,160,160,0,96,80,16,192,8,128,32,0
448  DATA 0,0,0,0,0,0,0,0,0,0,2,8,2,20,64,0,0,0,0,0,0,0,0
449  DATA 0,0,0,0,0,0,0,0,0,16,66,8,162,148,65,10,0,0,0,0,0,0,0,0
450  DATA 0,0,0,0,0,0,0,0,0,0,16,0,160,72,0,32,0,0,0,0,0,0,0,0
451  DATA 0,0,0,0,0,0,0,0,9,36,0,82,5,74,5,72,0,0,0,0,0,0,0,0
452  DATA 0,0,0,0,0,0,0,0,33,146,73,148,17,162,84,2,0,0,0,0,0,0,0,0
453  DATA 0,0,0,0,0,0,0,0,20,128,48,138,80,21,136,70,0,0,0,0,0,0,0,0
454  DATA 0,0,0,0,0,0,0,0,0,36,2,17,2,0,0,0,0,0,0,0,0,0,0,0
455  DATA 0,0,0,0,0,0,0,0,136,69,42,85,170,21,74,148,0,0,0,0,0,0,0,0
456  DATA 0,0,0,0,0,0,0,0,136,4,162,16,128,0,128,0,0,0,0,0,0,0,0,0
457  DATA 0,0,1,1,0,1,0,192,0,0,0,0,0,0,0,0,0,1,1,0,1,0,192
458  DATA 0,0,0,0,231,24,60,82,0,0,0,0,0,0,24,0,0,0,0,231,24,36,126
459  DATA 0,0,128,128,0,128,0,3,0,0,0,0,0,0,0,0,0,128,128,0,128,0,3
460  DATA 48,12,3,0,1,1,0,0,0,0,0,0,1,1,0,0,48,12,3,0,0,0,0,0
461  DATA 66,231,153,129,153,36,36,24,24,0,0,0,0,24,24,0,126,189,24,129,153,36,3
6,24
462  DATA 12,48,192,0,128,128,0,0,0,0,0,0,128,128,0,0,12,48,192,0,0,0,0,0
463  DATA 0,0,24,36,36,24,0,0,0,0,0,0,24,24,0,0,0,0,0,24,36,36,24,0,0
464  DATA 4,6,1,195,50,13,4,8,0,0,0,1,0,0,1,9,4,6,1,195,51,13,4,0
465  DATA 32,96,128,67,76,176,32,16,0,0,0,0,0,0,128,144,32,96,128,195,204,176,32
,0
466  DATA 0,0,1,1,0,1,0,192,0,0,0,0,0,0,0,0,0,1,1,0,1,0,192
467  DATA 0,0,0,0,231,24,60,82,0,0,0,0,0,0,24,0,0,0,0,231,24,36,126
468  DATA 0,0,128,128,0,128,0,3,0,0,0,0,0,0,0,0,0,128,128,0,128,0,3
469  DATA 48,12,3,0,1,1,0,0,0,0,0,0,1,1,0,0,48,12,3,0,0,0,0,0
470  DATA 66,231,153,129,129,0,0,0,24,0,0,0,0,0,0,0,126,189,24,129,129,0,0,0
471  DATA 12,48,192,0,128,128,0,0,0,0,0,0,128,128,0,0,12,48,192,0,0,0,0,0
472  DATA 0,16,0,16,16,16,16,0,0,16,16,16,16,56,124,0,0,16,0,16,16,16,16,0
473  DATA 0,20,34,8,34,20,0,0,42,85,42,93,42,85,42,0,42,85,34,73,34,85,42,0
474  DATA 20,0,85,0,85,0,20,0,20,42,85,42,85,42,20,0,20,42,85,34,85,42,20,0
475  DATA 0,231,231,88,80,102,126,24,231,24,24,167,46,153,129,102,0,231,167,0,8,
30,70,0
476  DATA 0,231,255,90,66,126,126,24,231,24,0,165,60,129,129,102,0,165,165,0,24,
0,66,0
477  DATA 0,231,231,48,16,78,126,28,231,24,24,199,110,177,129,98,0,231,231,0,32,
126,102,0
478  DATA 25,255,231,24,28,126,254,134,230,0,0,231,98,129,1,120,0,225,255,0,0,10
6,10,0
479  DATA 0,255,255,72,76,126,254,4,231,0,0,167,50,129,1,122,24,231,199,0,0,58,5
8,0
480  DATA 60,239,227,12,28,126,126,99,195,16,16,227,98,129,129,28,0,67,175,0,0,9
0,92,0
481  DATA 0,0,0,24,16,0,60,78,0,0,0,0,8,24,0,48,0,0,0,0,8,24,0,0
482  DATA 0,0,0,12,8,0,30,14,0,0,0,0,4,12,0,48,0,0,0,0,4,12,0,0
483  DATA 0,0,0,0,0,0,6,28,0,0,0,0,0,0,56,34,0,0,0,0,0,0,0,2
484  DATA 74,75,64,28,36,34,35,48,48,48,0,0,0,0,0,0,0,24,0,0,0,0,0
485  DATA 10,10,2,25,20,36,68,102,48,48,0,0,0,0,0,0,0,24,0,0,0,0,0
486  DATA 57,127,9,105,105,105,102,112,6,0,0,0,0,0,0,6,6,0,112,0,0,0,0,0
487 LOCATE10,10:PRINT "INPUT YOUR LEVEL"
488 LOCATE10,12:PRINT "HIGH - 1"
489 LOCATE10,14:PRINT "MID  - 2"
490 LOCATE10,16:PRINT "LOW  - 3"
491 LOCATE10,20:INPUT "INPUT NO.=";LEVEL
492 IF LEVEL>3 OR LEVEL<1 THEN BEEP:GOTO 491
493 LEVEL=LEVEL-1:LB=LEVEL*10:CLS:RETURN
```

移植のテクニック PARTⅡ

空丹 遊歩
(SORANI UFO)

N-BASICとの違い

NECのPCシリーズで作られた数多くのプログラム・ソフトをMZ-700で活用しよう，というのが今回のテーマです。

ハードウェアの面でも同じZ-80Aを使って4MHzで動いていますし，Hu-BASICは，N-BASIC，N₈₀-BASICとほとんど命令が共通していますから，BASICで書かれたビジネスプログラムやゲームプログラムは少しの変更で〝共有〟できます。

もっとも，MZ-700にはPCの80字(桁)モードに対応する方法はありません。しかし，40字20行，40字25行については移植可能なプログラムが多いのです。

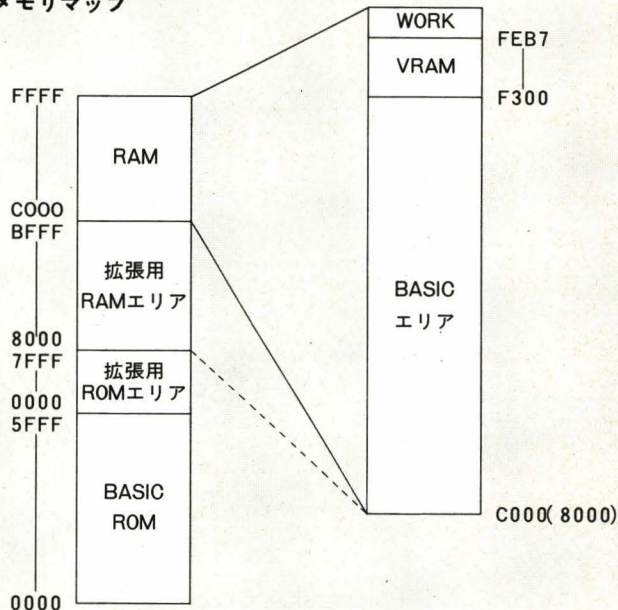
そこでまず，移植作戦の第1段階として，N-BASICで書かれたプログラムで，MZ-700Hu-BASICの命令と異なる部分から説明しようと思います。

表1をご覧ください。MZとPCの共通するコマンドは省略して説明してあります。

表1　PC-8001のキャラクタコード表、メモリマップ、コマンド

キャラクタコード表

＼	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		D_E		0	@	P		p	▁	┴		ー	タ	ミ	═	╳
1	S_H	D_1	!	1	A	Q	a	q	▂	┬	。	ア	チ	ム	╞	円
2	S_X	D_2	!!	2	B	R	b	r	▃	┤	「	イ	ツ	メ	╪	年
3	E_X	D_3	#	3	C	S	c	s	▄	├	」	ウ	テ	モ	╡	月
4	E_T	D_4	$	4	D	T	d	t	▅	▔	、	エ	ト	ヤ	◢	日
5	E_Q	N_K	%	5	E	U	e	u	▆	─	・	オ	ナ	ユ	◣	時
6	A_K	S_N	&	6	F	V	f	v	▇	│	ヲ	カ	ニ	ヨ	◥	分
7	B_L	E_B	▼	7	G	W	g	w	█	▕	ァ	キ	ヌ	ラ	◤	秒
8	B_S	C_N	(	8	H	X	h	x	▏	┌	ィ	ク	ネ	リ	♠	
9	H_T	E_M	)	9	I	Y	i	y	▎	┐	ゥ	ケ	ノ	ル	♥	
A	L_F	S_B	*	:	J	Z	j	z	▍	└	ェ	コ	ハ	レ	♦	
B	H_M	E_C	+	;	K	[	k	{	▌	┘	ォ	サ	ヒ	ロ	♣	
C	C_L	→	,	<	L	¥	l	¦	▋	╭	ャ	シ	フ	ワ	●	
D	C_R	←	−	=	M	]	m	}	▊	╮	ュ	ス	ヘ	ン	○	
E	S_O	↑	.	>	N	^	n	～	▉	╰	ョ	セ	ホ	゛	╱	
F	S_I	↓	/	?	O	_	o		┼	╯	ッ	ソ	マ	゜	╲	

CHR$

機能は同じですが，キャラクタとコードが違います。上記コード表を参照して(　)内のコードを変換してください。

COLOR A,B,C

カラーコードは共通ですが，Bは消したときのキャラクタを，またCはグラフィックスィッチを設定します。通常B，Cをカットして移植します。

CONSOLE A,B,C,D

A，Bは共通です。C，Dは0，40に変えてください。

DEF USR

機械語を置くメモリアドレスが違っていますので，それぞれのメモリマップを調べてください。

GET@/PUT@

グラフィックキャラクタを読み込んだり，表示したりする命令です。MZ-700にはこの命令がないのでキャラクタでカバーする以外にありません。

INP

装置とポートナンバーが違っています。キャッチするために使います。通常のゲームプログラムの移植であれば，INKEY$などで代用できるケースが多い。

LINE A,B

Aは行，Bは機能を決定し，Bの0または4がノーマル，1または5がブリング，2または6がリバース，3または7がリバースブリンクを指定しています。MZ-700では文字単位にCOLOR指定し，バックの色をかえて代用します。

LINE(X_1，Y_1)−(X_2，Y_2)，"A"，B

X_1，Y_1とX_2，Y_2を対角とする四角形をキャラクタで描きます(この場合はAの文字)。BFがついていますと枠の内側もキャラクタで埋めてしまいます。BやBFがついていない場合は対角をキャラクタで結びます。

これをMZ-700用に変更する場合，PSETで線や箱を描く以外にありません。

LINE(X1, Y1)－(X2, Y2), PSET, カラーコード, B

PSETまたはPRESETがつき，PSETで線を描き，PRESETで線を消します。グラフィックの大きさはキャラクタを横に4分割，縦に2分割した大きさですから，ディスプレートコード表(OWNER'S MANUAL 211)のF行で，また別のキャラクタで表現タで表現してください。カラーコードは同じです。B，B下またはこれらがついていないときの機能は前のラインキャラクタと同じです。もしLINEでゲームキャラクタが作られていたら，MZ-700の豊富なキャラクタの組み合せで対処します。この場合，GET@/PUT@が使われていますが，合成したキャラクタを文字列に入れておき，PRINT文で置き換えます。

LOCATE

40字モードは同一ですが，ビジネスプログラムなどで，80字モードのときは2行に表現するか，全く別な表現方法を考えねばなりません。一工夫が必要なところです。

USR

DEF USRと同じです。メモリマップを作って比較してください。

WIDTH

初期設定のなかでWIDTH80, 25となっていたら80字モードですから移植しにくくなりますが，40，25または40，20でしたら無視してかまいません。PCのCONSOLEとWIDTHはMZ-700ではCONS. だけで表わせます。

DATA表

```
N BASIC
INP(N) ノ データ

K=INP(N)
IF K=M THEN --

N     M        キーノ シュルイ
0 - 254   -   0 ノ キー
0 - 253   -   1 ノ キー
0 - 251   -   2 ノ キー
0 - 247   -   3 ノ キー
0 - 239   -   4 ノ キー
0 - 223   -   5 ノ キー
0 - 191   -   6 ノ キー
0 - 127   -   7 ノ キー
1 - 254   -   8 ノ キー
1 - 253   -   9 ノ キー
1 - 223   -   , ノ キー
1 - 191   -   . ノ キー
9 - 191   -   スペース キー

K=INP(9)
IF K=191 THEN --

モシ スペースキーヲ オシタトキハ
K ガ 191 ニ ナリマス
```

途中で投げ出さないこと

表1に紹介した以外の命令は共通です。

移植の第1歩はできるだけ短かなプログラムを選びだすこと，途中で投げ出さないことが必要です。途中で断念してしまっては面白いゲームがいつまでもあなたのものになりません。もう少しが大変なのです。それでは第2段階の説明に入ります。

PEEKとPOKE

動きは同じなのにこれほど手間のかかる命令はないでしょう。PEEKはメモリから内容を読み出す命令ですが，ゲームの場合，V-RAM(ビデオーラム)内を指定し，キャラクタの動きやゲーム進行をコントロールするために使っています。

PCの場合もX，Yの番地をV-RAMの番地に代入し，PEEKで読み出していますが，MZとは番地が違っているわけですから，PEEK，POKE文を使ったゲームは少し手間がかかるわけです。

MZ-700にはCHARACTER$(X，Y)があってPCほどやっかいではないのですが，PEEK，POKE文をマスターすると移植できるプログラムがかなり増えますから，張り切って進みましょう。

C＝PEEK(&HF302)

9F　C＝199　THEN～

とこんな形で使われています。メモリの番地は&Hがついて，16進数で表わします。

PCビデオラム

10-2-16シンスウ

```
   10-2-16 ｼﾝ ｽｳ

 10         2  16

  0  00000000  00
  1  00000001  01
  2  00000010  02
  3  00000011  03
  4  00000100  04
  5  00000101  05
  6  00000110  06
  7  00000111  07
  8  00001000  08
  9  00001001  09
 10  00001010  0A
 11  00001011  0B
 12  00001100  0C
 13  00001101  0D
 14  00001110  0E
 15  00001111  0F
 16  00010000  10
 17  00010001  11
 18  00010010  12
 19  00010011  13
 20  00010100  14
```

16進数はマシン語の勉強に不可欠ですから初めての方も覚えるようにしましょう。

ところでこのメモリには，PCではキャラクタコード表のコード，MZ-700にはディスプレーコード表(OWNER'S MANUAL 211，212ページ）のコードが書き込まれています。たとえばMZ-700の左上にUFOを表示するには

POKE　&HD000，&HC7 [↵]

とします。読み出しはPEEKです。

PRINT　PEEK(&HD000) [↵]

で「199」が出てきます。10進数の199は16進数でC7となりますね。

PRINT　&HC7 [↵]

とすると「199」が出てきます。これでおわかりのように，キャラクタは16進で書かれてもメモリ内には2進で書かれ，読み出しは10進です。ややこしいですね。

PRINT　CHARACTER$(0，0) [↵]

こんどは数字でなく，UFOがそのまま表示されました。

A$=CHARACTER$(X，Y)

9F　A$=〃　〃 THEN～

としますと，これはもちろん「もしX，Yの位置に〃　〃が描かれていたら～せよ」

10-2-16 シンスウプログラム

```
100 '10-2-16 ｼﾝ ﾌﾟﾛｸﾞﾗﾑ
110 '
120 CLS
130 PRINT" 10          2  16"
140 PRINT
150 FOR A=0 TO 255
160 B$="0000000"+BIN$(A)
170 B$=RIGHT$(B$,8)
180 C$="0"+HEX$(A)
190 C$=RIGHT$(C$,2)
200 PRINT USING"###    &        &  &&";A,B$
(A),C$
210 NEXT
```

MZ DISPLAYコード

MSD(上位) LSD(下位)	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111
0 0000	SP	P	0		→	↑	π		↓	セ	ワ	ム		日		SP
1 0001	A	Q	1		♠	<	!		チ	タ	ヌ	」	↓	月		
2 0010	B	R	2			[	"		コ	ス	フ	ィ	↑	火		
3 0011	C	S	3		■	♥	#		ソ	ト	ア	ュ	→	水		
4 0100	D	T	4		♦	]	$		シ	カ	ウ	ヲ	←	木		
5 0101	E	U	5		←	@	%		イ	ナ	エ	、		金		
6 0110	F	V	6		♣		&		ハ	ヒ	オ	ゥ		土		
7 0111	G	W	7		●	>	'		キ	テ	ヤ	ョ		生		
8 1000	H	X	8		○		(		ク	サ	ユ	°		年		
9 1001	I	Y	9		?	\	)		ニ	ン	ヨ	.		時		
A 1010	J	Z	-				+		マ	ツ	ホ	ェ	大	分		
B 1011	K		=				*		ノ	ロ	ヘ	ッ		秒		
C 1100	L		;						リ	ケ	レ	゛		円		
D 1101	M		/						モ	「	メ	。		¥		
E 1110	N		.						ミ	ァ	ル	ォ		£		
F 1111	O		,		:				ラ	ャ	ネ	ー	☺			

となるわけです。

このような場合，PEEKかPOKEのどちらかを使って移植します。

MZ-700のキャラクタビデオラム表とPCキャラクタビデオラム表を比較してください。PCはカラーデータ用としてアトリビューエリア40バイト＊25を，MZ-700は別表のカラーデータビデオラムをもっています。

移植の場合，PCのアトリビューエリアを無視してしまいますと，かなり1対1に対応します。F300から偶数番地（80字モードは奇数番地も使います)で，FE8Eまでが(アトリビューエリアを別にして)MZ-700

のD000からD3E7に大きなグラフ用紙に対応する番地を書き込むか変換プログラムを作るかして試してみるとよいでしょう。これが完成すればPEEK, POKEはもう恐くありません。

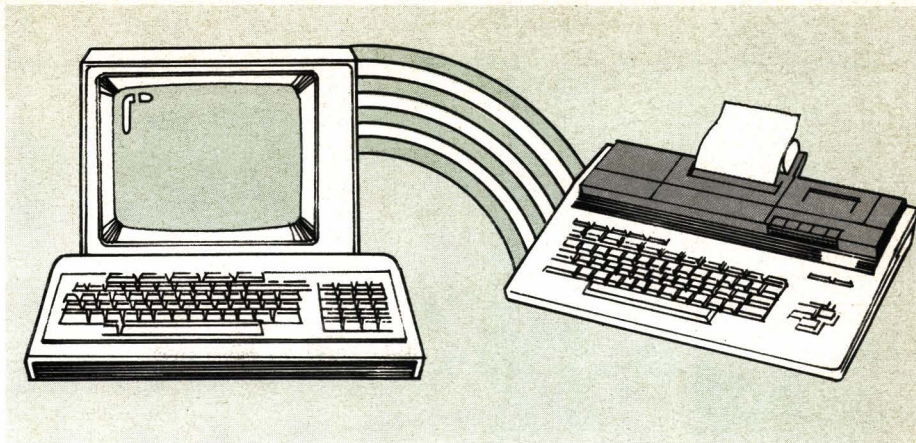
カラーデータ処理はMZ-700の方がかなり豊かです。

まず対応するキャラクタビデオラムにディスプレーコードを書き，ビデオラム番地＋2048番地にカラーデータを入れます。

PRINT HEX$ (&H キャラクタラム番地＋2048) ↵

または計算式をプログラムに加えて動きまわるキャラクタに色をつけていきます。

第3段は仕上げのカラー

ビデオカラーテストプログラムを動かしてください。360行の〝'〟(アポストロフィ)をとるとゆっくりと変化し，しばらくして入力メッセージが表示されます。8ビットのうち最上位ビットは英小文字とひらかなに変化させるビットで，画面上のAの位置，Nは何も関係ないビットです。CHRはキャラクタの色を，BACKはバックのカラーをコントロールするビットです。それぞれに光の3原色であるR(赤)，G(緑)，B(青)があり，どのビットを1にするかでカラーが決まります。ゲームプログラムは16進または10進でPOKEしてください。

このプログラムはビット単位の変化をみるためのもので，POKEするカラーデータがつかめると思います。2進入力ですが，16進，10進の変換をしています。ゲームプログラムは16進または10進でPOKEしてください。エンドレスですから SHIFT ＋ BREAKで止めるまで動いていますよ。

移植プログラムのグレードアップ

MZ-700は表に書かれた他にもう1画面分のキャラクタとカラーデータラムをもっています。そしてロールアップ，ロールダウンが強力です。これを使って，オリジナルをグレードアップしたプログラムに仕上げてみようとは思いませんか。もしオリジナルより素晴しいプログラムができたらそれはもうあなたのプログラムと言ってよいでしょう。

カラーテスト

```
100 'ビデオ ラム　　カラー テスト
110 '
120 CLS
130 COLOR 7
140 LOCATE 7,3:PRINT"カラー テスト"
150 LOCATE 21,12:PRINT"GRB GRB"
160 LOCATE 21,14:PRINT"CHR BAC"
170 COLOR 0,7
180 LOCATE 20,12:PRINT"A"
190 LOCATE 24,12:PRINT"N"
200 '
210 COLOR 7,0
220 FOR A=&HD000 TO &HD00F
230 POKE(A),&H1
240 NEXT
250 '
260 FOR A=0 TO 255
270 LOCATE 10,5:PRINT"ビデオ ラムニ カク カラー コード";A
280 LOCATE 7,7:PRINT"カラーコードノ 2 シン ヒョウジ"
290 A$=BIN$(A)
300 A$="0000000"+A$
310 A$=RIGHT$(A$,8)
320 LOCATE 20,10:PRINT A$
330 '
340 FOR B=&HD800 TO &HD827
350 POKE(B),A
360 'FOR T=0 TO 50:NEXT
370 NEXT:NEXT
380 '
390 LOCATE 20,16:PRINT"00000000      "
400 LOCATE 0,14:PRINT"ヒツヨウナ ビットヲ 1 ニ シテクダサイ"
410 LOCATE 18,16:INPUT A$
420 A=VAL("&B"+A$)
430 IF A>255 THEN 390
440 A$="0"+HEX$(A)
450 A$=RIGHT$(A$,2)
460 LOCATE 23,18:PRINT"  "
470 LOCATE 17,18:PRINT"16 シン ";A$
480 LOCATE 14,20:PRINT"     "
490 LOCATE 17,20:PRINT"10 シン ";A
500 '
510 FOR B=&HD800 TO &HD827
520 POKE(B),A
530 NEXT
540 GOTO 390
```

表　N-BASIC ビデオラム計算式

A 1

40字モード　モノクロモード

P=&HF300+X＊2+Y＊120

PEEK(P)

例　X=20：Y=10

P=&HF300+X＊2+Y＊120

PRINT HEX$(P)

F7D8がビデオラムの番地になります。

カラーモード

P=&HF302+＊2+Y＊120

PEEEK(P)

例　X=20：Y=10

P=&HF302+X＊2+Y＊120

PRINT HEX$(P)

F7DA がビデオラムの番地になり2バイトずれます。

カラーモードではビデオラムの一番左側の番地1列は使いませんので2バイトずれが起ります。

A 2

MZ−700

ビデオラム計算式

P=&HD000+X+Y＊40

PEEK(P)

Q=P+2048

PEEK(Q)

P はキャラクターラムの番地，Q はカラーデーターの番地です。

例　X=20：Y=10

P=&HD000+X+Y＊40

PRINT HEX$(P)

PRINT HEX$(P+2048)

DIA 4 がキャラクタラム

D9A4がカラーデーターラムの番地となります。

MZま表の他にもう1画面分のキャラクターとカラーのラムを持っています。

A 3

番地がマイナスになります。

PRINT D9A4　図

○−9820となり番地がマイナスになってしまいますす。では

PRINT BIN$(&HD9A4)図

○1101100110100100

2進にしてみますと上記の通りです。2進の最上位ビットが1の場合10進ではマイナス表示されますが，マイナス番地があるわけではありません。

PRINT HEX$(55716)　図

○D9A4 となり

PRINT BIN$(55716)　図

も上記と同じ2進になります。

アドレスの場合は10進の−9820も55716も同じ意味を持ちます。

日本ソフトバンク発行の雑誌

THE SOFT BANK 6号

6月30日発売予定――

○機種別/ジャンル別ソフトウェアリスト

パソコンソフト7000本満載

機能比較―ビジネスソフト

簡易言語/ワープロ/顧客管理/財務会計
給与計算/販売・仕入・在庫管理

ゲームソフト紹介

IBM・マルチステーションコンピュータ情報誌

Oh!55 創刊

好評発売中

定価480円

- 5550ハードウェアの全容
- カラー紹介/5550の内部をのぞく
- 5550ソフトウェア体系・概説
- インタビュー/5550の設計思想と展開
- 日本語BASICトラの巻
- 16bit機/ベンチマークテスト
- MS・DOSとCP/M86の比較
- 日本語文書プログラム(J・PACK 1)
- Multi plan紹介
- OAとはどうあるべきか
- Pascal/FORTRAN紹介
- マクロアセンブラMASM8086入門
- ゲーム・プログラム・リスト
  - スペースエンパイヤ
  - 宇宙大作戦
  - 万里の長城

愛読者プレゼント

1 MZ LOGO　MZ-2000用

モニター10名募集

パーソナル・コンピューティングの新しい時代をひらくインタプリタ言語として，最近とみに注目を浴びているLOGO（MZ-2000用，詳細なマニュアル付き，テープバージョン）をモニタープレゼントします。BASICと同等以上の演算機能に加え，強力なグラフィックス機能をもつ，人工知能言語「LOGO」に関心のある方，クリーンコンピュータを生かしてみませんか。　価格未定

2 東海クリエイト　簡易言語

S・CALGO　MZ-700用

モニター5名募集

MZ-700用に開発されたビジネスソフトを5名の方にモニタプレゼントします。

従来の簡易言語のように複雑な操作コマンドを覚える必要のないニュータイプのビジネスソフトです。演算，挿入，削除，表，ファイリング，検索，ソート，プリントなど，各種の処理機能がファンクションキーだけですべて可能。家庭，事務所，商店，学校のパーソナルOAとして威力を発揮します。　¥9,800

3 西日本マイコンセンター

㋐スーパーダイアード　3名

MZ-80B／2000用

㋑暗記訓練　3名

MZ-80K／C／1200用

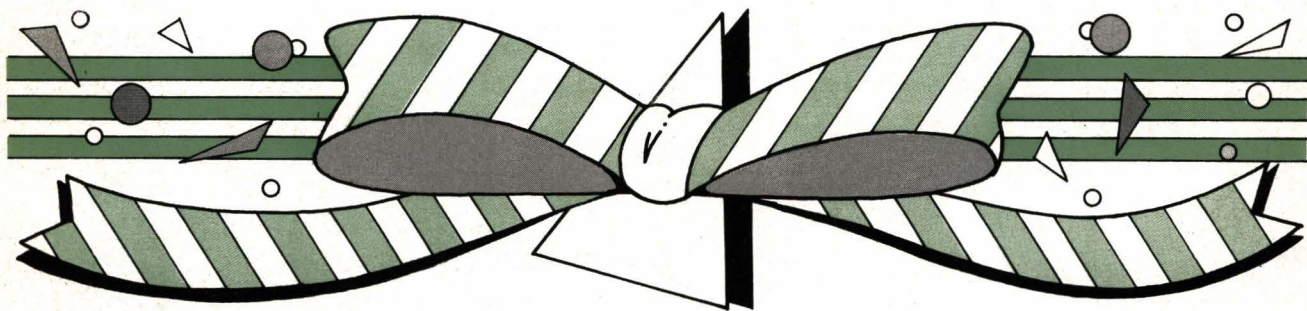
㋐使い方は自由，かつ無限です。調査項目は8項目までですが，確認したいことをYES，NO方式でインプットすると，8方向のグラフとなって表示され，傾向，情勢が一目でわかるもので，教育，経営分析，心理学，アンケートなど，応用範囲の広いソフトです。　¥8,800

㋑自己啓発ソフトとして，多方面に応用できます。あなたのMZ-80K／C／1200を記憶カード代わりに使って，学校の成績向上やビジネス用の知識情報のアップにご使用ください。　¥8,800

4 マイクロ データ ベース アソシエイツ

4人ポーカー

㋐MZ-80B／2000用　10名
㋑X1用　10名

あなたと3人分担当のコンピュータとの対戦です。持ち点300点で，誰かがゲームに参加できなくなると終わりです。さて，あなたは，勝つ前にどれだけゲームに参加できるかな？　各¥2,800

5月号プレゼント当選者発表

①PCG-700（大分県）江藤博昭（岩手県）高橋昭彦（福岡県）国友章　**②文筆**（世田谷区）濱津孝（和歌山県）塩崎正雄（埼玉県）枦元健一　**③アートクリエーター**（台東区）田島玲（静岡県）杉浦浩二（東村山市）高橋道明（石川県）倉田今希（奈良県）岡澤輝彦（兵庫県）武内敬次（岐阜県）鈴木昇（西多摩郡）井上英治（徳島県）野島隆（千葉県）小林隆司　**スカイダイビング**（埼玉県）稲妻正実（三鷹市）吉田和之（千葉県）本田善明（長崎県）谷山直司（茨城県）加藤明（岡山県）大多和研児（岐阜県）吉田匡輝（愛知県）戸崎宗（江東区）諸原新吾（荒川区）三木庸一　**④Oh!MZ Tシャツ**（広島県）波志登（愛知県）奥村晃弘（千葉県）石川淳一（長野県）田中正志（目黒区）前田勇人（神奈川県）高橋鉄也（長野県）吉川正規（大阪府）直井信也（長野県）高森今朝男（兵庫県）阪上佳秀（宮城県）荒川洋安（神奈川県）石亀健一（千葉県）倉島孝幸（岩手県）加藤剛（広島県）音丸誠司（石川県）衛良樹（和歌山県）辻本隆彦（愛知県）足立幸信（埼玉県）佐藤元浩（静岡県）市川博康　**⑤Oh!MZシール**（兵庫県）高橋良夫（千葉県）神田悟志（札幌市）藤谷研一（大阪府）安松谷高広（帯広市）後藤敏美（日野市）佐藤実（兵庫県）井澤裕明（滋賀県）吉川正司（愛知県）成瀬直幸（新潟県）伊藤丈晴　様
ほか90名様

以上の方々が当選されました。おめでとうございます。さっそくプレゼントの品を送らせていただきます。

プレゼントの応募方法

とじ込みのアンケートはがきに，希望のプレゼント番号を1つ朱記（3－アなど）して，7月10日までにお申し込みください。

訂正とお詫び

6月号「テクノソフト」のプレゼントソフト，「キャラクターメーカー3」の価格が¥2,000となっていましたが，¥2,500の誤まりです。

ぼくらの掲示板

売ります

★MZ-721。使用2ヶ月新品同様を5万円で。手渡し希望。少しおまけあり。〒151 東京都渋谷区西原2—21—10 三晃ハイム203 小西和彦

★売ります。MZ-721+N社製14型カラーディスプレイ+接続コード+ソフト3点を11万円前後で，昨年12月に購入したものです。W〒待っています。〒335 埼玉県蕨市塚越5—5—10逢來荘 牧田一成

★MZ-731+カラーTV100K円で県内なら，その後BASICも教えます。〒632 奈良県天理市櫟本町1115県住49—4 横山史郎

★売る。MZ-731（新品同様，保証付）を9万で。〒569 大阪府高槻市大塚町2—48—4 大村一

★MZ-2000，ゲームソフト+関連図書，58年2月購入を10万円で売ります。新品同様キズなし。近県は持っていきます。〒501-37岐阜県美濃市生櫛595 伊吹拓寛

★売ります。ツクモのMZ-700専用JOYスティック(新品)を3000円で。W〒ください。〒552 大阪市港区夕凪1—9—26明成マンション302 増本義彦

★PC-1500+CE-150，プログラム集2冊と他関連図書3冊を60K円で売ります（新品同様・保証期間内）まずは往復ハガキで，気長に待つ。近在の方はTELを。〒603 京都市北区紫竹牛若町28 ☎075-492-6129 狩野喜昭

★売ります。MZ-1200+保証書+ゲームソフト+マニュアル+BASICテープ他を7万円位で。少しくらいなら価格応談。送料はこちらでもちます。〒599-02大阪府泉南郡阪南町新町211 高松雅貴

★MZ-2000売る。本体+マニュアル+MZ-2000（及び80B）の関連図書を付けて，10万円で。配達の場合は10万円+α 格安のため早い者の勝ち。〒160 東京都新宿区北新宿1—25—20丸隆荘 松浦邦彦

★売ります。MZ-80B，MZ-8BG，GP-80DB，Hu-GBASIC，マニュアル類。無キズ完動品。150Kで。なるべく手渡し望む。〒184 東京都小金井市前原町5—5—16 ☎0423—81—5573 成松 洋

★MZ-700用簡易言語NEW-VIPテープ3本セット操作マニュアル320ページ付5月2日購入新品同様 8千円で売る。〒673 神戸市西区枝吉3—29（長崎の鐘内）渡辺秀雄 ☎078-929-2206㈹

★売る。PC-8001(32K)+PCG8100+ジョイスティック+ゲームソフト100数種を70K位で。〒428-03静岡県榛原郡中川根町藤川724 ☎05475—7—2313 山口源忠

★FX-9000Pを8万円くらいで売ります。まずは，あなたの希望価格を明記して往復はがきで!! 〒899-61 鹿児島県姶良郡吉松町川西873 ☎099575—3302 TELは20:00～21:00のあいだにしてください。九山年夫

買います

★PC-1251を求む！（安価か交換で，交換についてはリストを送ります）まずはハガキで。〒543 大阪市天王寺区味原本町五—十二 西弘悦

★MZ-2000（G-RAM1,2,3）を12万円以下で買います。しばらく待ちます。他にMZ-80K/C用ニデコカラーアダプタ（NH-MZD2）+ソフトを3万円以下で求む。まずはW〒で。〒723-01 広島県三原市新倉町6—23 勝部良次郎

★買います!! MZ-1200を20K円以下で！ 〒待つ 〒602 京都市上京区一条通り千本東入ル ☎075—451—8778奥川方 宇佐美真司

★求む。MZ-731を55K円で，又はMZ-721を36K円で。多少のキズは可（但し無改造の物に限ります）。送料はこちらで持ちます。くわしくはTELまたは〒で。☎02567—6—2874 〒953 新潟県西蒲原郡巻町添山81町 酒井孝和。

★拡張ユニットMZ-1U01を2万円で求む。〒560 豊中市末広町3—1—22 渡部透

★買いたし！ MZ-80Bと1ディスクを低価格にてお願いしたい。〒135 江東区東陽2-5-18-602 津金伸幸

★求む。MZ-700用カラープロッタプリンタを￥15000くらいで，又はMZ-700用システムキャリングセースを￥10000くらいで。〒862 熊本県熊本市保田窪本町608-1☎0963-82-3489（TELはPM7:00～9:00）福浦隆生

★買います。Sharp2000文字グリーンディスプレイを5千円で買います。送料貴方負担。手紙を待つ。〒960 福島県福島市南沢又字東谷地4—17植田光樹

★買います。MZ-K／1200／700シリーズを4万円でたのみます。売ってくれる方はテレビベーダー，ゲームウォッチ，ベーシックマガジン9月号から12月号をあげます。☎01233—3—4419 〒061-14 北海道恵庭市文京町116—21 富田勝裕 TELまってる。

★買います!! MZ-700用のカラーディスプレイ（MZ-1D05）+JOY-700を1万5千円で。MZ-1D05だけなら1万3千円～1万4千円。JOY-700だけなら2千円で。送料はこちら持ち。手紙で連絡を!! 〒085 釧路市南大通り3—3—3 佐藤崇 長く待ちます。

★求む。MZ-2000のG-RAM123を55K程度で。W〒長く待つ。〒319-14茨城県日立市田尻町905—63 遠藤泰まで。よろしくおねがいします。

★MZ1-R02をできるだけ安価で…。とりあえず往復はがきで。〒500 岐阜市元宮町4—6 ☎0582-51-0787 笠井義幸

交換

★こちらMZ-700+専用Gモニター+10万円，あなたCZ-800C+800D+グラフィックRAM（CZ-8GR）MZ-700は新品同様（3ヶ月使用）です。箱つき。〒639-11大和郡山市矢田山町55-3☎07435-5-0190酒井俊

★当方，MZ-2000。貴方，CZ-800Cを交換して下さい。できれば，グラフィックRMMかI/Oポートをつけて。連絡はW〒にて。〒371 群馬県前橋市大手町3—17—24 鈴木秀一

★当方，CASIO FX-702P。貴方，PC—1251／1250／1245又は，FXを1万5千円～2万円程度で売ります。TEL or 〒を…。〒733 広島市西区南観音5丁目4—17 ☎082—232—7769 沓内慎知

仲　間

★MZ-731を持っている高1の女の子，またはきょうだいがMZ-731を持っている高1の女の子，MZ-731のプリンターを使って文通しましょう。まずは手紙で。〒367 埼玉県本庄市久々宇99 内島俊実 高校1年生で～す。

★今回 全日本〝パソコン：X1〞愛好会を発足しました。会費は無料！ X1に関する情報交換等をアクティブに運用します。会員以外の方の連絡も大歓迎！連絡はW〒で待っています。〒316 日立市東金沢町4—6—3 萩庭彰

★MZ-2000を持っている人。買ったばかりの人や長くやっている人など，どしどしハガキか電話をください。くわしくは，ハガキに書くか電話しますので。とにかく，さいこうの仲間がそろっています。遠くでも近くでもかまいません。2000を持っている人は友達だ。☎07443—3—5234 〒636-03奈良県磯城郡田原本町笠形208 県営住宅5—12 水本全彦

★MZファンのみなさんMZ-1200復活クラブをつくりませんか。いま失われつつある1200を復活させよう!! 〒762香川県坂出市谷町1-9☎08774-6-2276 宮井健志

その他

★フリープレイ・スペースでマイコンと遊ぼう！ X1，MZ-700，PC-8801など設置。ディスク，テープもめちゃ安だよ！ 一度，遊びにおいで。待ってるヨ!〔アイ・オー・ライダーズ〕大阪市南区難波千日前15—17 サカイビル北館3F ☎06—641—2535

★練馬にOPEN!! 気の弱いあなた，家にばかりいないで一度遊びに来ませんか！ とにかく頭のリフレッシュに平和台に集合！ パソコンショップ&スクール〔マイコンシティ平和台〕〒176 練馬区平和台4—10—4 第2三晴ビル6F ☎03—937—1102

◆掲載ご希望の方は，とじ込みハガキに項目（売る・求む…）を明記してお申し込みください。なお，連絡先住所・氏名等も行内にお書き込みください。

◆内容は特に限定しませんが，ハガキの注意書きに反するものは一切掲載いたしません。

◆取り引きについては当編集部では責任を負いかねます。当事者間でお願いします。

◆原稿到着順に順次掲載していく予定ですが，応募者多数の場合，掲載できない場合もあります。ご了承ください。

Oh!MZ 質・問・箱

Q **MZ-80Bのユーザーです。あのMZ-2000の16ビットボードキットMZ-1M01は80Bに使用できるのですか。**

兵庫県　山岡　仁

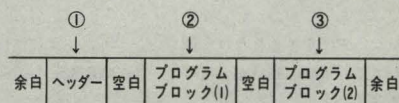
A 残念ながら16ビットボードキットはMZ-2000用となっており，80Bでは使用できません。理由として，ボードの装着するスペースの問題，そしてサポートするBASIC16（ソフトウェア）が問題となります。80Bの場合，16ビットボードキットを仮に外付けとしてもサポートするBASICがMZ-2000用となっており（MZ-2000の使用を前提として開発されている），2000のBASICが80Bで起動しないというのと同じことです。この16ビットボードキットは，新製品MZ-2200には使用できます。また，80B用の同ボードの発売は今のところ未定とのことです。早く開発してほしいですね。

Q **ぼくはMZ-721を持っていますが，最近SP-5030のソフトを買いました。それをMZにロードしてみました。ロード終了後テープレコーダにかけてみたところまだ続いてプログラムが入っているような音がします。途中で止まってしまったみたいですが正常でしょうか。**

大分県　古門哲安

A MZ-700やMZ-2000などは，プログラムをセーブするときプログラム本体を2回続けてセーブしています。これは不慮の事故から大切なプログラムを守るために行なっているためです。下図にカセットテープに記録する様式の概略を示しましたが，①の部分が壊れてしまいますと，②，③が正常であったとしても，そのプログラムは読めなくなってしまいます。図の①の部分（ヘッダー）は，そのプログラムのあらゆる情報（ファイル名，プログラムサイズなど）が書き込まれています。

②の部分が壊れているとMZは③の頭の所まで空読みして，この③をロードします。②，③の双方が壊れている場合も①が壊れているのと同様ロードすることができません。内田さんの場合は，第1ブロックでロードがうまくいっていると思われますので正常です。使っているうちに，ロード時間が長くなって来たなと思われたら，前述した理由により，オーディオカセットテープが寿命となることも考えられますので，バックアップを取っておくことをおすすめします。

	①↓		②↓		③↓	
余白	ヘッダー	空白	プログラムブロック(1)	空白	プログラムブロック(2)	余白

Q **シャープBASIC　SP-5030についておたずねします。SP-5030とSP-5030V1.0Aとの違いを教えてください。MZ-80K用のプログラムでPOKE文付近で動かない場合があります。また，オートリピート機能をいちいち入力するのも面倒です。POKE$3D25，0：USR(33)：USR(36)としてみたのですが，オートリピート機能付きのバックアップコピーが作れません。何か良い方法はありませんか。**

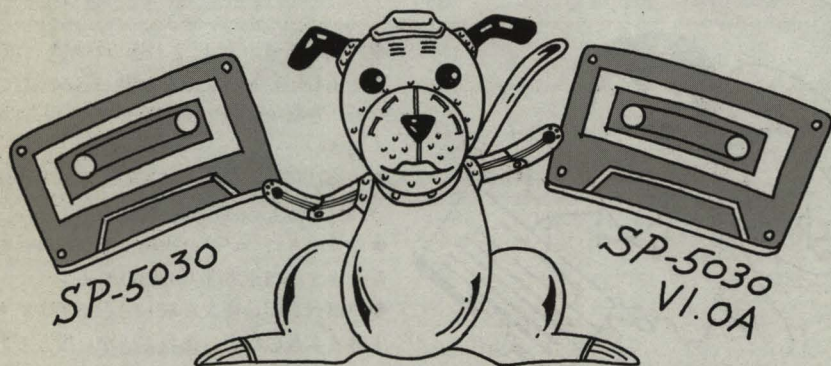
兵庫県　内田　隆

A SP-5030V1.0Aは，SP-5030の改良版です。すなわち，デバッグをほどこしたものですので，基本的には違いはありません。しかし，BASICインタープリタ内部（もしくはワークエリア）をPOKE文で書き変えて走るプログラムは走らない場合もあります。これは，デバッグのためにBASIC内部のアドレスがずれてしまったために起こる現象で，特記に値するような得策はなく，BASIC内部を解析し，POKEする内容を検討するしか方法はありません。

さて，オートリピート付のBASICを作りたいとのことですが，残念ながらこれではできません。これは，BASICがコールドスタートをしますと3D25，1としてオートリピートをオフにしてしまうからです。

Q **MZ-80Bを使っています。近くプリンタを購入しようと思いますが，シャープ製以外のプリンタ（たとえばエプソンのFP-80など）は接続できるのですか。教えてください。**

北海道　松田竜雄

A 以前本誌でシャープ製プリンタ以外はおすすめできないとの解答をしたと思います。話が重複しますが，本体と周辺機器とを接続する場合，インタフェイスが必要となってきます。プリンタの場合，一般的にセントロニクス社準拠といわれる8ビットコードにより情報を伝送する手段が使われていますが，シャープのMZシリーズに関しては，このセントロ仕様の出力となっていません。また，MZ用にシャープから用意されている各種ドットプリンタ

もセントロ仕様でないため，他のコンピュータには絶対接続できません。

しかし，他のプリンタ，松田さんが考えてらっしゃるFP-80は，現在市販されているプリンタの中で最も高性能であるため，広告などを見ていても出来たらMZに接続したいと考える方も多いのではないでしょうか。

結論から申し上げて，MZ-80Bおよび2000/2200にセントロ準拠のプリンタを接続することができます。ただし，2点ほどの問題点が生じますので，この辺を充分留意されてから入手されるのが良いでしょう。

① キャラジェネが異なる…MZ特有のアルファベット反転文字やグラフィックキャラクタが，セントロ仕様プリンタ側が持っていないため，化けて印字される。

② オブジェクトレベルでのプリンタ制御に制限される。

接続するためには，シャープ製のプリンタI/Fではセントロ準拠信号を出力できませんので，(株)I・Oデータ機器から発売されているプリンタI/Fボード（型名PIO-3050，ケーブル付属）を介して接続してください。

くどいようですが，シャープ製純正プリンタに比べて，かなりの制約が生じますので充分留意してください。I/Fボードは各種プリンタ用に何種類かありますので購入するときに指定する必要があります。

Q **MZ-2000に家庭用カラーテレビを接続することはできますか。**

広島県　山本　仁

A 家庭用テレビ受像機をコンピュータに接続するためにはRFモジュレータという一種のアダプターを使用します。MZ-700はこのRFモジュレータを本体に内蔵していますから直結することができますが，MZ-2000は，RGB出力しかもっていません。

NECや富士通など他社から，RFモジュレータが市販されていますが，それらは，自社仕様のため，MZ-2000用としては使えません。シャープからもMZ-2000用としてRFモジュレータは発売されていませんから家庭用TVの接続はできません。

Q **MZ-2000のユーザーです。本体購入時に付属していたBASICテープ(MZ-1Z001)は起動時に“V 1.0b”と表示されるものですが，マシン語データ領域に書き込む場合アドレス$FED0～$FEFFの間にバグがあるみたいです。データを書き込みテープにセーブし，再びロードしなおすと$FED0～$FEFFの間ではセーブしたときのデータと違っています。なぜでしょうか。**

岡山県　原田一郎

A MZ-80B/2000のBASICは，起動するとBASICテキストエリアの先頭からFEFFまでを00でクリアしたあと，FF00番地にスタックポインタを設定してコマンド待ちになります。

このスタックポインタというのは，BASIC内部のDATAエリアの格納に使われていて，通常FEDO～FEFF番地付近を使用しています。

今回のような原因は，このスタックポインタが内部データを書きかえているためであり，BASICのバグではないものと思われます。

マシン語DATAエリアを設定するときは必ずLIMIT文を実行してください。

Q **X1で打ち込んだプログラムをテープにセーブし，MZ-731に読み込ませて同機のプリンタでリストをプリントアウトすることは可能でしょうか。MZ-731で走らせることは考えず，“虫取り用”に使用できれば良いのですが。よろしくお願いします。**

埼玉県　清水　宏

A X1とMZ-700シリーズの双方ともHuBASICがありますが，同じHuBASICといってもテープにセーブするボーレートという速度が異なります。MZ-700は1200ボーですが，X1は約2倍の2700ボーとなっています。このため，MZ-700でX1上で作成したプログラムを直接読むことはできません。ただし，自分でこのボーレート変更のプログラムをオブジェクトレベルで作成すれば可能です。しかし，中間コードが異なりますので，各機種特有の命令については保証されませんし，キャラジェネも一部異なるため印字が化けてしまいますから，リスト出力，デバッグなども同一のコンピュータで行なった方が得策であるといえそうです。

質問にお答えします

日頃疑問に思っていること，どんなことでも結構です。どんどんお便りください。難問，奇問，編集部が総力をあげてお答えいたします。ただし，お寄せいただいているものの中には，マニュアルを読めばすぐに解答が得られるようなものも多々あります。最低限，マニュアルは熟読しておきましょう。質問はなるべく具体的に，こと細かに書いてください。返信用切手同封の質問をよく受けますが，原則として，質問には本誌上でお答えすることになっていますのでご了承ください。

宛先：〒102 東京都千代田区四番町2-1
(株)日本ソフトバンク
出版部「Oh! MZ質問箱」係

from READER 読者から編集室へ to EDITOR

●広告が少ない。　浅井　正平（34）新潟市

♥多いほうがよいというのでは？

●初心者にもわかりやすくて、とってもよい雑誌だと思う。Oh!MZ5月号を買うために回った本屋及びマイコンショップは5軒をこえました。　大西　通雄（35）奈良市

♥5月号は売り切れ店続出でご迷惑をおかけしました。おつかれさま！

●まだ事務処理用プログラム紹介が少ない。最近特に内容が充実してきてうれしい。　松本　次男（30）富田林市

♥次号では、簡易言語など、ビジネスユースに関するものにも力を入れる予定です。期待してください。

●MZ−700の記事が増えてきてうれしい。ゲームプログラムの場合、プログラムの説明をした方がいいのではないかと思う。　桶口　篤志（16）北海道深川市

●Oh!MZスバラシイ!!　他のマイコン誌がアホに見える。書店にさっそく定期購読を予約しました。Oh!シリーズ以外は書店にて立ち読みで終りでしょう。　飯山　幸助（30）北海道浦河町

♥過激な発言！

●内容にはマンゾクしているが、初めてのパソコンで、プログラムを貴誌のリスト通り入れて、動くには動くが内容がわからず、他への応用がしにくい。プログラムの解説をもっといれてほしい。せめてREM文をわかりやすく。　山北　裕（31）滋賀県甲賀郡

♥意味もわからずに、長いリストを打ち込むのは、はっきり言って苦痛でしかありません。そこで使われているテクニックなどを理解しながら打ち込めるような解説も必要でしょうね。

●プログラムや編集がとてもよかった。いい雑誌だと思う。意見といえば、機種の紹介がもっとほしい。　藤井　秀行（11）青森県八戸市

♥とくにこれからパソコンを買う予定の人は、各機種の性能や個性を知りたいことでしょう。

●MZの専用の雑誌ということで非常に助かっています。私は教職の仕事に就いておりますので、成績等の管理がほとんどですが、その点に関する記事（マシン語による高速処理等）もあればと思います。ただ素人のため、専門用語ばかりで構成されている記事には頭を痛めております。もっとわかりやすくお願い致します。　吉垣　隆雄（28）大阪府河内長野市

♥できる限りわかりやすく表現するよう気をつけます。少しずつ勉強していってください。

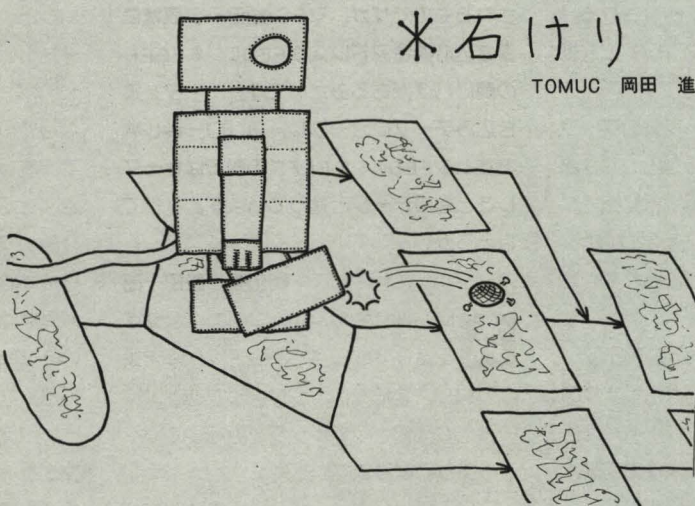
●5月号のX1の記事ちょっとへったんでない？　X1を可愛がってください。ゲーム特集はよかった。　武市　耕司（17）北海道美明市

●貴誌のような機種別（メーカー別）のものは、今、貴重な存在です！　私が思うに、アニメ講座があるのだから、CG関係のほうにも力を入れていただきたく思います。　河村　功治（16）秋田市

●MZ−2000のグラフィックプログラムなどを、もっと載せてほしい。アニメーションなんかもっともっとほしいですねー。　和田　忠（51）京都市

♥今やパソコンもグラフィックの時代です。視覚的にも楽しめるようなアイデアがあれば紹介していきたいと思います。

＊石けり

TOMUC　岡田　進

●5月号は感動的なものがなかったようである。中途半端なまとめ方で相変らず原稿を羅列しているだけとしか思えないが、こんなものだろうという気もする。　関口　光英（22）埼玉県大里郡

●非常に見やすくなったと思いますが、MZインタビューとか、息抜きのようなものが減ったようで、内容に堅苦しさを感じます。　田中　秀敏（24）大阪市

♥できるだけ楽しく読んでいただけるよう心がけているのですが、内容を充実させようとするあまり、少々かたいものになったかもしれません。専門誌であることにこだわって、排他的な雑誌にならないよう気をつけたいと思っています。

●くやしい！　一週間がかりでスーパーゴルフを入力して、バグをとり、改良を重ねてやっと18ホールまで遊べるようになったと思ったら、完全なリストが載った。　太田　稔（24）宮城県泉市

♥ほんとうにご迷惑をおかけしました。ゴメンナサイ。

●MZ−700を所有していますが、MZ系専用雑誌として、非常に便利であります。マシン語のサブルーチンが特によくわかります。　清水　威夫（26）奈良市

♥Oh!MZが誕生してから約一年、読者のレベルの向上には私たちもおどろいています。

●市販ソフトの改良（BASICの機能の付けたしなど）は参考になりました。パソコン情報誌にBASIC講座なんていらない。特にHu−BASICを数ページで語れないと思う。　辻本　隆彦（22）和歌山市

●初心者向けの記事も多く、パソコンを始めたばかりの人のためには良いと思うが、BASICばかりなので不満である。FORM、PASCAL、LISPなどのプログラムの掲載を求む。　深井　克志（21）新潟市

♥いろいろな言語が使えるはMZの最大の魅力です。来月号では、みなさん待望のLOGOが登場の予定です。乞うご期待。

●いろいろ本を買い、見くらべましたが、表紙やグラビアの色がとてもキレイです。字の大きさも適当で、とても見やすいです。本によっては、ただプログラムだけあり、なんの説明もないとか、プログラムの使い方が十分ではない……などで、初心者の私としてはいつもたいへんな思いをしてました。Oh!MZはその点、とても親切に書かれているので気にいっています。　根津　祐子（25）山梨県富士吉田市

●僕たちの高校で、今年度からマイコン同好会が発足。そしてなんと、Oh!MZが会の推せん誌になりました。これからも楽しいプログラムを待っています。　秋田県雄物川高校マイコン同好会一同

♥なんとうれしいお便り。編集部一同、推せん誌の名に恥じないようにがんばりマッス。

★5月号アンケートはがきより掲載

●編集部から読者の皆さんへ

お便りをお寄せください。内容はどんなものでも構いません。言いたい放題、ユニークな内容のものをお待ちしています。あわせて、イラストも大募集。「Oh!MZ」読者から編集室へまでどんどんお送りください。

from READER 読者から編集室へ to EDITOR

●前からほしかったこの本は、5月号でやっと買えました。ぼくはMZファンですので、関係のないX1、2000の記事もよかったです。これから毎月買おうと思います。
梶原　直樹（13）泉市

●8月号から購入してますが、内容が充実してきたと思う。しかし、SHARPの機種も多くなってきたので、全部を取り上げるには、ページ数が不足のような気がします。これからも内容の濃い本にしていってください。
小山　孝広（25）福岡市

♥7月にはMZ－2200、秋にはPC－5000が発売されるとのこと、世間の期待とは裏腹に、ユーザーの皆さんは心おだやかではいられないことでしょう。本誌Oh!MZでは、質・量ともにパワーアップを計り、各機種ユーザーの要望に答えたいと思っています。

●ビギナーの私にもわかるような内容や、高度な内容をごちゃまぜにしたので毎月買っていきたくなるような本ですね。今回のようにゲームだけでなく、それをつくる手だけ…そういうものはビギナーにとってはありがたいかぎりです。
細畑　隆行（16）石川県輪島市

♥アイデアをどういかすかが大きなポイントになるのです。

●パソコンの初心者にとって、ユーザーズマニュアルだけではとても使いこなせないため、このような機種別情報誌はたいへん助かる。
大野　豊（35）神奈川県茅ヶ崎市

♥パソコンを自分の目的に利用するには、応用性の高い実践的な内容が必要ですね。

●MZに関する本という事で、いつも全ページを見ています。MZ－80Bに使用できそうな物はないかと……あれば手帳に控えてプログラム作成に利用しようとしております。
真弓　幸則（33）三重県四日市市

♥自分の機種とちがう記事でも、興味を持って読めば、きっとなにかヒントが得られるのではないでしょうか。アイデアを貯えて、ぜひともいいプログラムを作ってください。

●5月号の「読者から編集室へ」でおさわがせしましたが、薬剤師の国家試験は無事合格しましたので安心して読めました。そこで「アソシアトロン」というのをうちこみました。機械が悩んでいるというのはなんともおもしろいものです。
畑中　範子（22）兵庫県西宮市

♥合格おめでとうございます!! これからもがんばってください。

●今回初めて、この本を買ったが、マイコンの楽しさがよくわかった。これからもがんばってほしい。
谷　直人（13）高知市

♥こういうおたよりが、なによりも励みとなるのです。ありがとー。

●色々あった（8色＋中間色）のでヨロシイ！　今の記事の配分はいいと思いますが、ヨクを言わせてもらいますと、見たい記事は、例えば、マイコン利用例○○会社、and今後のコンピュータ社会と人間の仕事の変化などについての対談○○氏vs○○氏など。最後にひとつ、BASIC以外の言語も少しは…。
佐藤　昭夫（28）西頸城郡

♥なかなか広角度な興味をお持ちのようですね。もっといろいろな分野に興味をもつ人が増えると、パソコン利用の可能性が広がると思うのですが。

●"マシン語・魔神語"もうちょっとやさしく書いてもらいたい。ついていけない。できれば、完結したら一冊にまとめてもらいたい。
近藤　吉浩（19）香川県高松市

●毎月買って読んでいる。良い記事が多いが、入門したての私にはBASIC以外入力できないようなので、マシン語の入れ方などをもっとやさしく説明してほしい。
渋沢　伸行（21）中野区

♥マシン語については、6月号、今月号の特集で、かなり勉強になったのではないかと、ひそかに自負しているのですが、いかがでしたでしょうか?

実力の「オール・イン・ワン」(K2E)。いわば「応用力の差」(80B)。磨きぬかれて新次元へ(2000)。気軽にスキンシップ(1200)。こんなMZを待っていた(700)。映像とコンピュータのクロスオーバー(X1)。
NEXT TITLE?

沼田　博好　京都府

●僕はこの前、本誌4月号のGALACTICA WARSを必死で入れていた。そして「あと6行だい!」となり、感動してふるえる指で入れていた時、ふいに画面に何も映らなくなった。見ると、な、なんとコンセントがぬけているではないか!!　それからの僕は、プログラムを入れる気力をなくしたのだった……。
賀谷　達樹（12）国分寺市

♥ほんと、なさけない気持ち、よくわかります。でも、そこは若さでもう一度チャレンジ！

●もっと筋のとおった企画があっていい。
沢田　高宏（17）名古屋市

♥ミもフタもない…。

●アンケートはがきに切手をはらなくてもいいのはうれしい。
吉田　勝久（19）富山県高岡市

♥あの～～、お願いしますよ。

●内容がとても充実してきたと思います。毎月発売日を首を長くして待っています。やはりマイコンはMZでよかったとユーザーを安心させ満足させる記事を期待しています。それとMZ－700のプロッタプリンタ応用を載せて下さい。
守田　富男（29）熊本県八代郡

♥4色のプロッタプリンタを使って編集室へお手紙をだそう…とか。

●4月・5月号あたりから、次号を「待つ」というかんじになってきたようです。それまでは、本屋にあれば「いちおう買う」ぐらいのかんじのしろものでしたよ。やっとMZの本が出たという印象を持てる本になってきました。SFマガジン風の表紙も気に入りました。今後は、初心者向けのシリーズでおちこぼれを作らないように進めて、中級者向けのシリーズを加えていって下さい。
高石　公資（29）兵庫県西宮市

♥初心者から、ある程度力のある人までをカバーするのは大変ですが、とにかくがんばります。

●主にMZ－700の記事しか読まなかったのですが、ゲーム作りに始まり、ビジネス用あり、逆アセンブラありで、MZ用雑誌だから当然でしょうが、他の雑誌では載るか否かのMZ－700の豊富な記事に満足しております。
鳥飼　紀雄（31）佐賀県鳥栖市

♥できる限りバラエティに富んだ誌面づくりをしていきたいと思っています。

●一週間前パソコンに興味を持ち始めて以来、本を13冊も買って勉強しています。テレビが壊れたという単純な理由でX1を買います。もう車はやめてパソコン一筋!!　今では夢の中にまで出てきます。
内田　洋一（20）千葉県船橋市

♥13冊!?　ウーン大変なおもいこみですね。でもX1はこわさないでくださいよ。

●ワーイ・ワーイ！　X1買ったぞー、うれしいな、うれしいな、ワーイ、ワーイ、バンザーイ、X1バンザーイ!!
大野　政治（15）大阪府門真市

♥ワーイ・ワーイ・バンザーイ！　Oh!MZバンザーイ!!　ん?　失礼致しました。

●4月号から購読しているNew faceですが、なんといってもMZの専門誌というのがうれしい。もっと早く気づけばよかった。
本間　敏明（20）札幌市

♥今後とも、どうぞヨロシク！

●Oh!PCに比較して非常に読みやすい記事が多く、PCファンの私も時々Oh!MZを購入して読みます。MZ－2000用16ビットボードキットが発売とのこと。MZファンはいいなあ！
西岡　昌一（41）神奈川県藤沢市

♥イヤー、このハガキ、PCの担当にも見せてあげたいですね－。

INFOMATION

ポケコンのニュータイプ

——PC-1253, PC-1253H——

シャープは，いつでも，どこでも，誰でも簡単につかえる電卓感覚のポケットコンピュータ「PC-1253」(RAM容量4.2Kバイト)と「PC-1253H」(同10Kバイト）の2機種を新発売しました。

必要なソフトウエアはPC-1251(RAM容量4.2Kバイト)かPC-1251(同10Kバイト)を使って開発しますが，このプログラムをPC-1253, PC-1253Hに入力し,実務処理するユニークなポケコンです。

操作はわかりやすいカタカナ表示やYES, NOの対話方式によってルーチン処理するもので，一度プログラムを入力すれば5年間もソフトの保存ができます。価格はオープンです。

マルチテレビジョン新発売

——カセットレコーダ内蔵——

シャープは，映像情報の新時代を前にして，ただ番組を楽しむだけでなく，ユーザーが情報を自由に選んで使いこなす「21型音声多重ショットビジョン」(21C-GIE) を発売しました。

① テレビにデータカセットレコーダを内蔵し，パソコンのデータレコーダとして使えるほか，カセットデッキとしてテレビ音声や外部入力音声をステレオ録音，再生ができます。

② 文字多重放送をはじめとする各種ニューメディア機器，ホームビデオ，ビデオディスクなどと合理的にインテグレートするよう，1) 21ピンRGBマルチ端子つき，3系統ビデオ入力端子を標準装備(40字×25行, 5×7ドット／字) 2) 文字，図形表示に適した直角平画 (SQ) ブラウン管の採用，3) 映像情報を忠実，鮮明に再現するくし形フィルター回路を内臓しています。

③ 多機能化するテレビ機能を正確にコントロールし，画面にわかりやすく表示するニューデジタルショットビジョンを内蔵しています。

④ 映像信号方式の異なるヨーロッパなどで生産されたビデオディスクの垂直振幅を自動補正します。価格は23万円です。

投稿●MZ-80B／2000 HuBASIC Ver.2.0

BACK UPコピーの作成

大阪府 唐沢 浩

還ってきたHuBASIC

HuBASICとはMZ-80K/C時代からのつきあいで，すみずみまで使い馴れていたBASICですから，MZ-2000に買い換えていらい，ずいぶん永い間さびしい思いをしていたものです。

この間，コマンドの少ないMZ-1Z001を相手に悪戦苦闘の連続で，たとえばPRINT USING命令をMZ-1Z001でどう表現するか，など悩まされたわけですが，それはそれで大いに勉強させられました。しかしMERGEやRENUMBERがないのにはホトホト弱り，スクリーン・エディット機能を利用して，テープを換えながら，ハンドリナンバー，ハンドマージする方法を開発？するなど，ここでも大いに勉強させられた次第です。

もっとも私がK/C用のHuBASICを購入したのもマシン語モニタ機能をもっているからで，K/C時代にはヒマをみつけては，BASICとモニタの間を往復しながら，CPUのカラクリを追跡し，ひたすらコンピュータの秘密を嗅ぎまわっておりました。

HuBASIC Ver2.0/2000によって，モニタやBASICの中身がどうなっているのか，また新しい楽しみが増えたと，ひとりほくそ笑んでいる私です。

ところで，HuBASIC／2000は高価なテープです。さっそくBACK UPテープを作ることにしました。

準備—— 2つに分ける

BACK UPの方法は,HuBASICでHuBASICを読む方法と，MZ-1Z001でHuBASICを読む方法の2通りが考えられますが，HuBASICの内部解析がまだ終っていないため，今回はMZ-1Z001を利用することにしました。

その方法ですが，まずHuBASICをIPLからLOADしてテープカウンタの数をメモします。私の場合は，

005 モニタ部のLOADING開始
019 モニタ部のLOADING終了
016 いったんREWIND後，BASIC部のLONDING開始
024 BASIC部をFOUNDしていったん停止し，すぐLOADINGを再開
086 BASIC部のLOADING終了して REWIND

以上，最初から最後まですべて自動的に行なわれますが，ここで，019→016へいったんREWINDするのは，HuBASICがモニタとBASIC部の2つのプログラムに分かれているからで，BACK UPコピーを作成するときも，この2つの部分を別々にコピーする必要があります。

モニタ部のコピー

① MZ-1Z001をIPLからLOADINGし，BASIC/MONITOR MANUALの205ページに記載された手順にしたがってモニタを変更します。続いてモニタコマンドLによってHuBASICモニタ部を続み込みます。モニタ部の続み込みが終了するとテープは自動停止し，画面はコマンド待ちに戻ります。

② 新しいカセットをセットし，モニタコマンドJによって01B5番地にジャンプし，いま続み込んだモニタ部をSAVEします。テープは自動停止し，画面はふたたびコマンド待ちとなります。

③ 念のため，SAVEしたHuBASICのモニ

ペンギン情報コーナー

「パスカル」誕生

――サプライ専門ショップ――

東京・秋葉原の専門店街に「サプライ専門ショップ　パスカル」(高山幸雄店長，東京都千代田区外神田　秋葉原ラジオ会館内)が誕生。取り扱い商品はフロッピーディスク，ペーパー，リボンなど，パソコンやワープロの活用に欠かせないサプライ・プロダクツが主体。「交換コーナー」を設置して，顧客ユーザーがハードウエア，ソフトウエアを自由に交換できるようにするなど，関連サービスも充実しています。

ポスタープレゼント

――FDイメージ　PARTII――

富士フイルム株式会社は，Oh!MZの読者30名に，同社フロッピーディスクのイメージポスター '83　PARTIIをプレゼントします。官製ハガキに，「イメージポスター希望」と明記し，住所，氏名，年齢，性別，電話番号，本誌名を併記して下記の宛先までお送りください。締切りは昭和58年7月25日（当日消印有効）です。

〒106 東京都港区西麻布2-26-30
富士フイルム株式会社　磁気材料部
「FDポスタープレゼント」Oh!MZ係

「Oh!FM」が出展

――パソコンFM祭り――

富士通(株)主催の「パソコンFM祭り」が東京・池袋サンシャインシティ文化会館4階で開かれます。FMシリーズのソフトおよび周辺機器などを取り扱っている約40社が出展します。

本誌の姉妹誌である「Oh!FM」も出展します。

会期は7月8日(金)から10日(日)の10時から18時までです。但し，10日は17時まで。

住所移転

ピーシーエー㈱東京本社は，東京都新宿区西新宿7-7-33　新銘ビル新館2F　電話03(368)9631㈹に移転しました。

タ部 IPLからLOADしてみましょう。

LOADが終ってモニタが走り出し，

HuBASIC　ロード　チュウ

＊L

の画面が出てテープが回り始めますので，カウンタ020で[SHIFT]+[BREAK]を押してテープを止めます。

BASIC部のBACK UP

④ 次はいよいよBASIC部です。

HuBASICは先ほどのテープカウンタを見ても判るとおり，MZ-1Z001の約2倍の容量があります。したがって，モニタ部と同じように&H8000番地以後に読み込む方法ではメモリが不足するのでLOADできません（暴走してしまいます!）。

⑤ そこでもう一度，IPLからMZ-1Z001をLOADして，&H7000番地から読み込めるようにモニタを書き換えます。

01E5	E9	→	F7
0289	2A	→	21
028A	54	→	00
02B9	2A	→	21
02BA	54	→	00
02BB	11	→	70

⑥ さて，準備ができたら，ふたたびHuBASICをセットし，テープカウンタを020にあわせると，BASIC部の頭出しができますから，モニタコマンドLでBASIC部をLOADします。

BACK UPの完成

⑦ BASIC部のLOADが終了したら，先ほどの新しいカセットをセットし，モニタコマンドのJで01B5番地にジャンプさせ，モニタ部コピーの後にSAVEすればBACK UPが完成です。

なお，MZ-2000のテープへの読み書きはきわめて正確で，エラーの発生は考えなくともよいため，モニタ部のみのIPL-LOADテストは省略できます。これによって手順はもう少し簡単になるでしょう。

ファンクションキー定義の変更

BACK UPコピーを作成するついでに，自分の好みにあわせてファンクションキーの初期設定を換えてみましょう。

ファンクションキーのデータは &H1094番地から &H112B 番地にストアされています。ダンプリストをとると，マシン語コードとともに，それに対応するAUTO,SCREENといった文字が出ますから，カーソルを動かしてそのままスクリーンエディットします。

データの形式は，最初の1バイトが文字数で2バイト目以降がキャラクタのマシンコードですから，キーの定義を換えたときは忘れずに文字数も書き直しておきましょう。またMZ-1Z001と違って文字列の最後の0Dは不要です（0Dは↲が必要なときに使用するのですが，↲のコードは7Fです）。

書き換えがすみましたら同じ番地を再度ダンプするか，BASICへ戻るかして，動作に間違いがないかチェックし，OKなら変更箇所の番地とコードをメモしておきます。これらの作業は，BACK UPの準備段階でHuBASICを走らせたとき，ついでにやっておきます。

さてこの &H1094～&H112B 番地は HuBASICのモニタ部にあり，&H7000番地以後に読み込まれるため，それぞれ &H8094～&H812Bに入ることになります。モニタ部をLOADし終ったらMZ-1Z001のMコマンドで &H8094から1バイトずつ書き込んでください。

この作業は，マシン語の経験がゼロの方でもASCIIコード表を片手に楽しくはかどります。意外なところでマシン語の面白さを発見するでしょう。

（編集部注：MZ-2000に関する投稿ですが，80Bの場合でも同様にBASIC SB-5520を使用すれば，BACK UPコピー，定義変更が可能です）

SHARPのブースにLOGOが登場した

ビジネスショウ／マイクロコンピュータショウ'83

新製品NEWS／PC-5000／MZ-2200／MZ LOGO

パソコンファンにとっては最大のイベントであるビジネスショウ，マイクロコンピュータショウですが，ことしも連日の大にぎわいで，ゆっくり観覧できず，ひたすら会場を駆けまわってカタログ資料をあつめるだけ，といった向きが多かったようです。

ビジネスショウでは相変らずワードプロセッサの人気が高く，50万円台を切る機種がシャープと富士通からあいついで発表され，「年内は30万円台か」と予想する声がいちだんと強まってくることでしょう。

マイクロコンピュータショウでは16ビットマシンへの関心が強かったようですが，いまのところパソコンのレベルで16ビットの機能をフルに発揮するハードウェアが用意されたとはいいがたく，ユーザーもまだまだとまどっているのではないでしょうか。

パソコンユーザーがBASIC（＋マシン語）オンリーの世界に限界を感じて，もう少し強力で，アプリケーションの範囲が広大なレベルに関心をもちはじめているのは事実です。従来の新製品紹介がメインのショウから，もう少しユーザーが自分のマシンの機能をアップできるシステムアプリケーションに馴染む機会としてのショウへ、転進がはかられるべきかもしれません。

ショウ光景

初めて50万円を切ったパーソナルワープロWD-800

新製品MZ-2200

MZ LOGOのデモ光景

LOGOの登場

ビジネスショウでは，SHARPから，16ビットCPUを搭載した1台3役のポータブルコンピュータ，PC-5000が披露され，ユーザーの熱い視線を浴びていたのが印象的です。1台3役というのは，16ビット，8ビットデュアルのパソコンとして使えるほか，日本語ワードプロセッサのシステム・アプリケーションをもっていること，RS-232Cインタフェイスを内蔵しており，データターミナルとして使えること，です。液晶表示のディスプレイですが，640×80ドットのグラフィックス，漢字で40字×4行，プログラム表示で80桁×8行と実用的で，BASICのほかにMS-DOS Ver 2.0が標準装備されるという画期的な携帯用コンピュータであり，パーソナルユースとして手軽に持ち運べる汎用コンピュータです。発売は10月からです。

マイコンショウの正面入口にブースを構えたSHARPは，MZ-2000の普及タイプとして，MZ-2200を出品していました。MZ-2000から，ディスプレイとデータレコーダ

を取り外したもので，ユーザーの必要にあわせてカラーディスプレイやディスクが使えるよう配慮してあります。BASICも従来のMZ-1Z001のほかにカラーBASIC MZ-1Z002の2本が標準装備され，G-RAM，拡張ユニットも実装されました。

注目したいのはLOGO言語の登場です。SHARPのほかに2，3のブースでLOGOのデモンストレーションが行なわれていましたが，いずれも黒山の人だかりで，ユーザーの関心の高さがうかがえました。しかし，どのブースも，LOGOの特長について満足に説明できる係員がいなかったのは残念です。

LOGO言語は簡単なプログラムで複雑な図形処理ができるタートルグラフィック機能のほか，再帰性がありますので合理的なプログラムがつくれますし，ユーザーが自分でコマンドを作成・定義でき，これらを積み重ねていっそう複雑なプログラムを組みあげることが可能です。何よりもBASICインタプリタ言語への徹底した批判と人工知能の研究を集約するかたちで誕生した言語ですから，LISPの特長をあわせもち，パソコンに推理・推論させることができるなど，明晰かつ単純なプログラミングで即実行するインタプリタとして，高い評価をうけています。

SHARPでは，クリーンコンピュータ設計を生かし，国内他社に先がけて本格的なLOGOを用意するそうです。MZ-2000／2200用のLOGOですが，発売時期は7月の中旬，価格は未定ですが2万円内になるとみられています。これによって本誌でもLOGOを用いた楽しい記事を提供できるでしょう。

注：MZ-2200の購入者には2000～3000円の特価でLOGO（テープバージョン）が提供される予定です。

16ビットHHC PC-5000発売！

先月号で速報しましたが，シャープから16ビットポータブルコンピュータPC-5000が発売されます。

このPC-5000は，1台でコンピュータ，日本語ワードプロセッサ，データターミナルの3役をこなす電池駆動の16ビット携帯用コンピュータです。

オフィスにおいても急速にOA化が進み，特に日本語ワードプロセッサの需要は増加

ポータブルコンピュータPC-5000

PC-5000の豊富な周辺機器

MZ-2200

の一途をたどっています。

大規模OAシステム導入の前段階としてPA（パーソナルオートメーション）が進む今日，個人が自由に操作できる汎用機は今後欠くことのできない機器になることが予想されます。

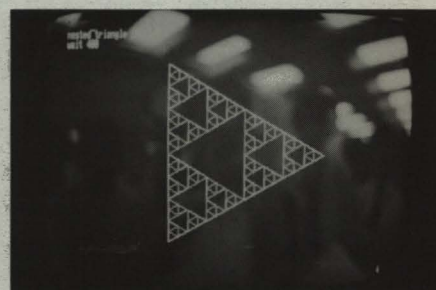
PC-5000は，本格的実務レベル用途として充分耐えられるよう設計されています。1台3役といっても各々の機能が簡易型であったとすれば，それこそ単用途（専用）機を必要に応じて揃えれば済みます。しかしできることならば，1台ですべての用途にも対応できる本格的な汎用機があればこれに越したことはありません。

（特徴）

1．電池駆動16ビットシステム

16ビットCPU（8088）と新開発のC-MOS LSI群の組み合せによるデュアルCPU（メイ

SHARPのブース

電磁メカ機構をもつデータレコーダとミニフロッピードライブ

LOGOのデモンストレーション

シャープポータブルコンピュータPC-5000

ン16ビット，サブ8ビット）を採用。低電力システム化により，コードレスで連続8時間の使用が可能。

2．高度なビジネス・アプリケーションへの対応

本体の主記憶用メモリ容量は標準128Kbytes実装（最大256Kbytesまで増設可能）しており，日本語ワードプロセッサ機能を始め，高度なビジネスアプリケーションソフトの搭載ができます。オプションとして128Kbytesのバブルカートリッジが用意されていますので，どこでも簡単にファイル管理が行なえます。

OSは，16ビットコンピュータの代表的ソフトウェア，MS-DOS（Ver 2.0）を採用。プログラム言語も，マイクロソフト社製BASICを搭載しています。MS-DOSバージョンで世界中に流通している数千本にもおよぶ，豊富なアプリケーション・ソフトウェアが利用できます。

MS-DOS管理下以外では日本語ワードプロセッサ機能を付加するユニットが，オプションで用意されています。これを実装すると，本格的な日本語ワードプロセッサとして使用できます。

PC-5000は，RS-232C I/Fを内蔵しており，各種コンピュータ間とのデータ交信（ターミナルとして使用）も可能です。

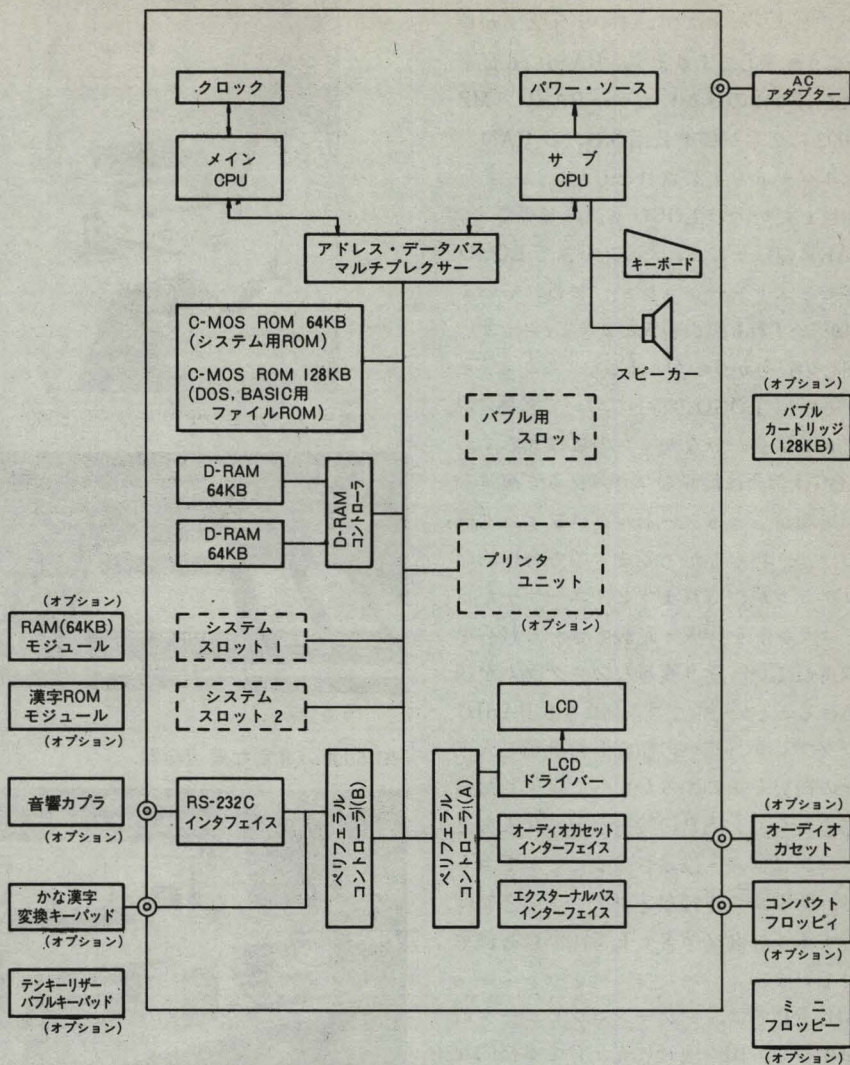
3．マンマシンインタフェイスの向上

ディスプレイは，先だってシャープが開発に成功した高密度大型液晶ディスプレイを採用し，一覧性のある表示（640×80ドット）を実現しています。また，携帯性を重視すると一番の問題となるキーボードも標準タイプライタキーボードを装備し，ディスクトップタイプと変わりない操作性を実現しています。

プリンタは，本体内に内蔵できる電池駆動熱転写式で，24×24ドットの漢字出力および単票A4サイズに印字ができます。

その他，各種周辺機器（フロッピーディスク，ディジーホイールプリンタ等）が接続できるようにインタフェイスを装備していますので，拡張性にもすぐれたパーソナルコンピュータです。本体価格 350,000円。今年10月発売予定です。

品　名	本　体	プリンタユニット	日本語ワードプロセッサユニット
標準価格	350,000円	80,000円	150,000円
発　売	10月		

PC-5000システム構成図

MZシリーズの新鋭機 MZ-2200登場！

シャープパーソナルコンピュータMZ-2200

シャープは，5月23日，パーソナル・コンピュータMZシリーズの主力機種としてMZ-2200を7月から発売すると発表しました。23日といいますと，本誌グラビアでも紹介していますがマイクロコンピュータショウ '83の前々日にあたり，滑りこみセーフといったところです。

このMZ-2200は，従来のMZ-2000用として市場流通ソフトウェア（約1,000本といわれる）や，その周辺機器もそのまま使える，いわゆるコンパチブル性に大きな特徴を持つほか，従来機（MZ-2000）ではオプションとなっていたカラー・グラフィック機能およびI/Oユニットが標準装備となり格段に充実した内容となっています。

外観は，先だって発表されベストセラーになったMZ-700シリーズと同様セパレートタイプとなり，グリーンCRT，データレコーダはオプションです。

新製品MZ-2200は，MZシリーズのメモリ空間オールRAM設計や操作性の良さ，高信頼性，シャープ製高速処理BASIC，拡張性を考えた設計など，基本の良さを生かし，今後多様化する用途に適応できるハイコストパフォーマンスモデルとして注目されます。

（概要）

◆ハードウェア

1．ディスプレイ，データレコーダを分離

したセパレートタイプ。

2．MZ-2000用オプショングラフィックボードおよびメモリ（MZ-1R01, MZ-1R02×2相当）が標準実装。

3．インタフェイスカードを装着するための拡張ユニット（MZ-1U01相当）を標準実装。

4．従来本体後部にあったIPL, RESETスイッチおよび音量ボリューム（スライド式）を前面に配置。

5．MZ-2000用オプション16ビットボードキッド（MZ-1M01）取り付け可能。

◆ソフトウェア

現行機種MZ-2000とフルコンパチブル。

今までMZ-2000や80Bでオプションとなっていたグラフィック関係，フロッピー，プリンタ等の接続に不可欠な拡張ユニットまでも本体購入時にすべて付属されていますので，拡張時にいちいち本体内部を開いて接続するといった手間が省けます。また，BASICはMZ-1Z001の標準タイプの他カラーテープBASIC（MZ-1Z002）までも付属するといったサービスぶり。確かに最近の機種はほとんどカラー対応となっていますから，コントロールするソフトウェアを付属することは評価できます。CRTについてもグリーン12インチを付けるよりユーザーレベルで自由に選択できることも喜ばしい限りです。

何といっても一番の魅力は価格128,000円で現在各社激戦価格帯に位置します。シャープでまともにグラフィックが扱えた機種はMZ-2000の218,000円でしたからこのMZ-2200の登場により，MZシリーズが身近なものとなりました。

ソフトウェアの互換性は，先ほども述べたとおりMZ-2000とフルコンパチブルです。すなわち，MZ-2000用として市販されているパッケージものあるいは，雑誌等の各種媒体に発表されたプログラムはすべて新製品MZ-2200上で動作できます。これは，特にハードウェア等の仕様変更がないためで，以前80B⟷2000の互換性についてかなりの制限が生じたのとは異なり，安心してMZ-2000用のソフトウェアが利用できます。

なお，MZ-2200の発売に伴ない周辺機器にも新製品が同時に発売されます。なかでもミニフロッピーディスク（MZ-1F07）はインタフェイスカードとケーブルが付属して158,000円というお買得。ドライブは上位に当たるMZ-3500シリーズと同じ薄型のものが使われているため場所を取りません。新規発売予定としてはこの他，漢字ROM，コンパクトフロッピー（日立系列規格のものと思われる），音響カプラー（DN-90E相当），そしてC-MOS拡張RAMなどがあり，MZシリーズのラインナップもより充実してくるものと思われます。

MZ-2200は今年7月中旬発売予定です。定格その他のスペックは，表を参照してください。

MZ-2200一般仕様

CPU	SHARP LH0080A（Z80A-CPU）
クロック	4MHz
メモリ	ROM 2Kバイト （IPL：イニシャル・プログラム・ローダ） ROM 2Kバイト （キャラクタ・ジェネレータ） RAM 64Kバイト （ダイナミックRAM） RAM 48Kバイト （グラフィックRAM）
画面表示	キャラクタ表示 8×8ドッドマドリックス 1）1,000キャラクタ （40キャラクタ×25行） 2）2,000キャラクタ （80キャラクタ×25行） 1),2）：ソフトで切り替え グラフィック表示 1）320×200ドット(標準BASIC) 2）640×200ドット(標準BASIC) 3）カラーコントロール(拡張BASIC) ページ1, 2, 3およびキャラクタとの混合表示が可能 1),2）：ソフトで切り替え 白黒時3画面，またはカラー時1画面
データレコーダインターフェイス	標準オーディオカセットテープ使用 データ転送速度 2,000ビット/秒 データ転送方式 SHARP PWM方式 マニュアルおよびソフトウェアによりコントロール

サウンド出力	最大 40mW（440Hz）
キー構成	キー数 92 ASCII標準，10数字キー ファンクションキー，カーソルコントロールキー カセットテープデッキコントロールキー
編集機能	カーソルコントロール：上，下，左，右， ホーム，クリアキー 挿入，削除キー
時計機能	内蔵
電源	AC100V±10% 50/60Hz 消費電力 約50W
温度	使用温度 0℃～35℃ 保存温度 －15℃～60℃
湿度	使用湿度 80%以下
重量	約7kg（拡張ユニット取付時8.5kg）
外形寸法	幅 440mm（拡張ユニット取付時 440mm） 奥行 490mm（562mm） 高さ 118mm（118mm）

拡張用オプション（新規発売分）

製品名	機種名	標準価格	発売時期
データレコーダ	MZ-1T02	19,800円	7月
シリアルプリンタ	MZ-1P07	79,800円	7月
ミニフロッピーディスクドライブ	MZ-1F07	158,000円	7月
プリンタインタフェイス	MZ-1E08	9,000円	7月
1P07用接続ケーブル	MZ-1C35	6,800円	7月
1P07用トラクタセット	MZ-6P09	8,800円	7月

MZ-2000／2200用に新システムソフト MZ LOGO発表！

先月末に開かれたマイコンショウ'83のシャープブースで黒山の人だかりであったものに，LOGOがありました。

LOGOは，現在最も多く使われているBASICに比べ，命令が自然言語（英語）だけでできていること，代数に関する知識が必要ないことから，初心者や子供を対象とした，教育・ホビー用の新しいプログラミング言語として期待されていますが，BASICを超える使いやすい高級言語（インタプリタ）として高い評価をうけています。

今回，MZ-2000／2200用に発表された「MZ LOGO」は，アメリカのUNISON WORLD INC. が開発したもので，APPLE LOGOと同様，テラピンLOGOのサブセットタイプにあたるようです。

現在のところ対象となる機種はMZ-2000/2200ですが，MZ-2200の発売時期にあわせてSHARPから販売されるとのことです。

SOFT TOUCH

商品名／アートクリエーター

分類／ユーティリティ
言語／BASIC
機種／CZ-800C(グラフィックRAM要)
媒体／カセットテープ
マニュアル／なし(プログラム中で説明)
価格／3,800円
発売元／マイクロ データ ベース アソシエイツ

X-1をコンピュータグラフィックマシンに変えるソフトウェアです。ちょっと大げさな表現ですが，PCG機能を生かしたいわゆるパターンエディタですが，15種のコマンド，色パターン自動生成機能など多機能で便利なツールです。とかく，PCG関係の定義は手間のかかるものですが，本ソフトを利用することにより，比較的容易に作成することができます。

1．四角，直線，円，楕円，多角形の描画機能。
2．筆の太さ調整，描画速度調整，ぼかし，かすれ等を描くブラシ機能。
3．640，320両モード対応。ジョイスティック可。
4．PALET変更機能，カセットテープへのデータLOAD，SAVE機能。
5．色パターンは自由に生成可能。

商品名／プログラムデバッガーSET
(MZ-1Z001用)

分類／ユーティリティ
言語／マシン語(モニタよりLOAD)
機種／MZ-2000/2200(MZ-1Z001用)
媒体／カセットテープ
価格／5,000円
発売元／㈱テクノソフト

```
*L
FILE NAME:
LOADING MZ-2000 DBG V1.0
Ready
/

 PROGRAM DEBUGER V1.0
 ── TECNO-SOFT TM. ──

   TRACER-ON      =(T)
   TRACER-OFF     =(O)
   RENUMBER       =(R)
   DELETE         =(D)
   APPEND         =(A)
   VARIABLE-DUMP=(V)
   CMD.SEARCHER  =(S)
   PSE COLLECT ??(

Ready
```

BASIC(MZ-1Z001) は，約21KBytesというコンパクトのため，ユーザーエリアが他に例を見ないほど大きく取れますが，BASIC上の制約が多く，特にデバック時には不便を感じます。

このプログラムデバッガーSETは，トレーサ，リナンバー，デリート，アペンド，変数ダンプ，コマンドサーチャー機能がBASIC上から制御できるようになり，プログラム開発時に最大の助手になってくれます。
※ディスクBASICでは使用不可。

商品名／スーパーC.R.Tコピー3
(BP5&P6専用)

分類／ユーティリティ
言語／マシン語(モニタよりLOAD)
機種／MZ-2000/2200共用
媒体／カセットテープ
価格／6,000円
発売元／㈱テクノソフト

BASICにハードコピー用の命令として，COPYコマンドがありますが，そのままではグラフィックとキャラクタを混在してプリンタに出力することができません。

このスーパーC.R.Tコピー3は，その欠点を補うユーティリティーソフトで，グラフィック，キャラクタの混在画面をプリンタ(MZ-80BP5，P6) に出力します。
通常のCOPY命令の大きさの他，一画面分をペーパー一枚分に引き伸ばしてコピーすることもできます。この場合，縦横比が1.03：1となり，仮に円を描いた時，ほぼ真円を出力できますので，グラフ出力の正確さを要する場合便利なツールといえるでしょう。

商品名／キャラクターメーカー3

分類／ユーティリティ
言語／BASIC

機種／MZ-2000/2200共用
媒体／カセットテープ
価格／2,500円
発売元／㈱テクノソフト

このプログラムは，前出のスーパーC.R.Tコピー3のキャラクタフォントを再定義するための専用ツールです。文字パターンは，このツールのパターンエディタによって作成します。汎用性を持たせるためにはキャラクタを1セット（256文字）定義する必要があります。

○MZ-80BP5，P6，GP-80DB用。
○このソフトウェア単独で使用することはできません。

商品名／UNDERGROUND CITY
（地下都市）
分類／反射神経型ゲーム
言語／マシン語(IPLからロード)
機種／MZ-2000/2200共用／カラー対応
グラフィックRAM3ページ分要
媒体／カセットテープ
価格／2,800円
発売元／シルバーボールソフトウェア㈱

本稿締切り直前に入手したソフトウェアで，アメリカのシルバーボールソフトウェアという会社が製作したゲームソフトです。

反射神経型ゲームとしては，かなり製作技術が高度で，グラフィック処理やサウンド効果も充分使いこなしている秀作ソフトウェアといえるでしょう。

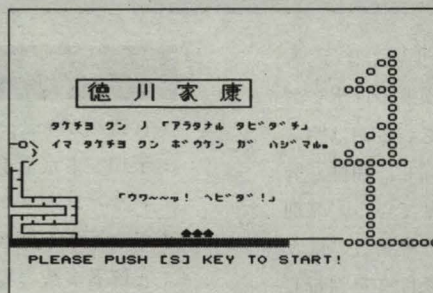
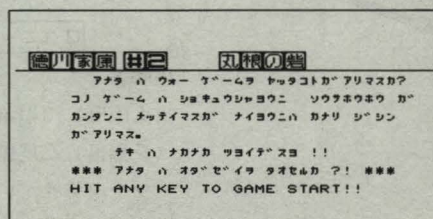
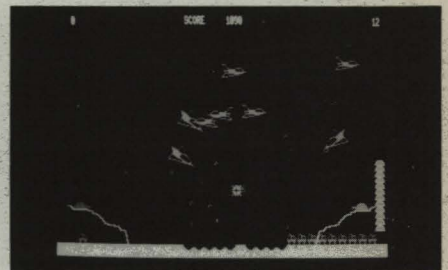
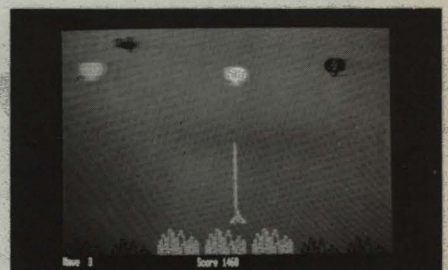
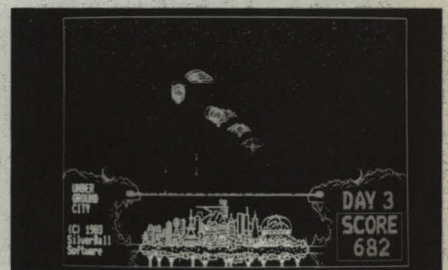
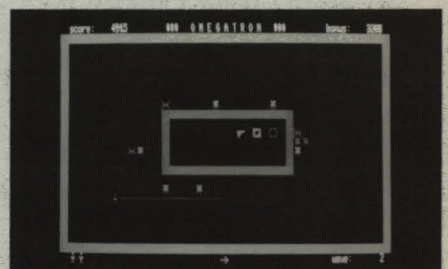
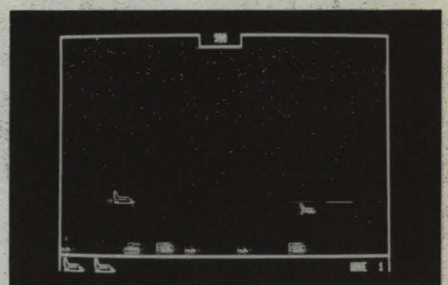
これは，Cコンパイラで記述したものをオブジェクトに落とし，IOCSを付けて，IPL起動の方式でカセットテープで供給されています。

今回日本に上陸したゲームソフトは，8本ですが，このUnderground Cityはなかでもグラフィックのきめ細かさに圧倒されてしまいます。

宇宙からの侵略から守るため，都市全体を地下にもぐらせたうえバリアーを張り，敵とたたかいます。バリアーが破られないようにして，地上の2つの基地から敵を攻撃します。敵の動きがとてもリアルで何か本当の生物を相手にしているような錯覚さえ感じられます。

最近は，ソフトウェアの質が問われる時代ですが，文句なし良質の分類に入るものといえるでしょう。価格も2,800円と手頃です。このゲーム以外に〝DAWN PATROL〞〝OMEGATRON〞〝MISSION〞〝FLYBOY〞（いずれもMZ-2000/2200用）が同時発売されます。右の画面写真を参照してください。

この欄に掲載をご希望の向きは，「Oh! MZ編集部SOFT紹介係」と朱記して，サンプルとマニュアルを下記あてにお送りください。
〒102　東京都千代田区四番町2－1
日本ソフトバンク出版部

商品名／徳川家康　①少年編
②桶狭間決戦前夜
分類／思考型ゲーム
言語／BASIC
機種／MZ-80B/B2/2000/2200共用
グラフィックRAMⅠ要
媒体／カセットテープ
価格／各4,000円(説明書付)
発売元／㈱ウス𛀁パソコンセンター

「連続大河ゲーム」と命名されたこのゲームは，5巻完結で各巻独自の内容となっています。〝連続〞ゲームたる所以は，各巻ともゲームを終了した時点でゲームの成績データをSAVEし，次のゲームを開始する前に，前回ゲームのDATAをLOADして続けるということにあります。そして，前回の成績次第で，次のゲームの難易度が大きく違ってきます。今回発表された2巻はそれぞれ，「ロールプレイング・ゲーム」，「ウォー・ゲーム」といったように，様々な形態をとっています。もちろん，単独でも遊べますが，シリーズとして揃えてみるのもよいでしょう。

SOFT TOUCH

MISSION

DAWN PATROL

UNDERGROUND CITY

OMEGA TRON

FLY BOY

COVER STORY〈4〉

夏はやはり，シーサイドで過ごしたい。というわけで，デッキチェアに似合いの，ライトな口当たりのSFを紹介しよう。

「アルタイルから来たイルカ」。イルカの歴史家が記録したストーリーという設定だ。人類とイルカは，百万年前アルタイルから地球に植民した人々の子孫というのだ。一方は，陸に上って人間に，他方は海に住んでSea people(と自称している)になった。

人類とイルカが兄弟なんて，想像しただけで楽しいね。イルカが，人間に捕われた同胞を助け出すために，海底峡谷に爆弾をしかけて，地震を引き起こすというプロットには，5月末の秋田沖地震のあとだけにドキッとしたけれど――舞台はカリフォルニア沖，ここと日本列島は同じ環太平洋火山帯に属していて地盤が不安定なのだ――全体としては，イルカと人間のファンタジックな交流がメインになっている。作者が女性（マーガレット・セント・クレア）で，しかも御主人が動物童話の作家というせいかもしれない。

一言付け加えると，クレア女史はアメリカでも，1940年代から活躍していたという女流ＳＦ作家の草分けである。（誠子）

7月号クイズ

今月は，演えき推理を必要とする論理パズルを出題しよう。

和雄と秀一と正は，ある飛行機の機長，副操縦士，ナビゲーターである。ところが誰がどの職種についているかは明らかでない。ところで，この飛行機には彼らと同じ，和雄，秀一，正という名をもつ3人の乗客が乗り合わせている。識別の便宜上，乗客のほうは〝……さん〞と呼び，乗務員のほうは呼びすてにすることにした。さて，以下の条件文を判断して，機長は誰かを答えてほしい。

1．正さんは神戸に住んでいる。
2．副操縦士は大阪市に住んでいる。
3．秀一さんは中学で習った代数をずっと以前にすっかり忘れてしまった。
4．副操縦士と同じ名の乗客は京都に住んでいる。
5．副操縦士と乗客のひとりである有名な数理物理学者は，ふだん同じサウナに通っている。
6．和雄はナビゲーターをポーカーで負かした。

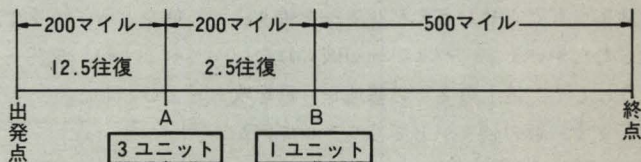
●応募方法●

官製ハガキに，答と住所・氏名・年齢を明記のうえ、
〒102 東京都千代田区四番町2－1
日本ソフトバンクOh！MZ編集室「7月号クイズ」係
までお送りください。締め切りは7月10日の到着分までです。

正解者多数の場合，抽選で10名の方にテンプレートを差し上げます。なお正解発表は9月号誌上で行ないます。

●5月号クイズの正解●

13ユニット(限界なし)

1ユニットで500マイルを走行できるのであるから，最終的には砂漠の終点より500マイルのところに1ユニットがあればよいことになる(図参照)。A点に1ユニット置くには，A，B間を2.5往復しなければならない。ということは，A点に3ユニット置く必要がある。A点に3ユニット置くには，出発点，A間を12.5往復しなければならず，これにより，出発点から持ち出すガソリン量は13ユニットと算出できる。

正解者

（大阪府）川端碩夫（愛知県）皆川洋喜（大阪府）宮脇周三（愛知県）高尾肇（石川県）高村幸夫様ほか5名の方にＴシャツを送らせていただきます。

ごめんなさいのページ

Oh!MZ 編集部

本誌のバグに関するお問い合わせは
受付電話　03(265)5789
受付時間　16：00～18：00

なお，お問い合わせは，原則として，本誌バグ情報のみに限らせていただいております。また，ご質問いただくときは，○月号○ページ，タイトルを明示してください。
また，2ヵ月前までのバグに関しては，このページでサポートできています。バックナンバーでお確めください。

4月号　PCG EDITORプログラム

P.62　右段7行目
「58030行のCG$をMIRROR$(CG$) とすると」はまちがいです。
「58015としてCG$＝MIRROR$(CG$) を追加すると」と訂正いたします。

5月号　マシン語・魔神語・DEUS EX MACHINA

P.66　リスト8と9を以下のとおり訂正いたします。

リスト8

```
01 0000                 ;
02 0000                 ; PROGRAM 4-U
03 0000                 ;
04 0000                 ;        UP ROLL
05 0000                 ;                     40*25  [80*25]
06 0000                 ;
07 0000                       REL   7000H
08 7000                 ;
09 7000 2100D0                LD    HL,D000H       ; HL=D000H
10 7003 110080                LD    DE,8000H       ; DE=8000H
11 7006 012700→28             LD    BC,0027H→28    ; BC=0027H→28   [ 004FH→50 ]
12 7009 EDB0                  LDIR
13 700B 1100D0                LD    DE,D000H       ; DE=D000H
14 700E 01C003                LD    BC,03C0H       ; BC=03C0H      [ 0780H ]
15 7011 EDB0                  LDIR
16 7013 210080                LD    HL,8000H       ; HL=8000H
17 7016 0E27→28               LD    C,27H→28       ; C=27H→28      [ 4FH→50 ]
18 7018 EDB0                  LDIR
19 701A C9                    RET                  ; RETURN
20 701B                 ;
21 701B                       END
```

リスト9

```
01 0000                 ;
02 0000                 ; PROGRAM 4-D
03 0000                 ;
04 0000                 ;        DOWN ROLL
05 0000                 ;                     40*25  [80*25]
06 0000                 ;
07 0000                       REL   7000H
08 7000                 ;
09 7000 21E7D3                LD    HL,D3E7H       ; HL=D3E7H      [ D7CFH ]
10 7003 112780                LD    DE,8027H       ; DE=8027H      [ 804FH ]
11 7006 012700→28             LD    BC,0027H→28    ; BC=0027H→28   [ 004FH→50 ]
12 7009 EDB8                  LDDR
13 700B 11E7D3                LD    DE,D3E7H       ; DE=D3E7H      [ D7CFH ]
14 700E 01C003                LD    BC,03C0H       ; BC=03C0H      [ 0780H ]
15 7011 EDB8                  LDDR
16 7013 2127B0                LD    HL,8027H       ; HL=8027H      [ 804FH ]
17 7016 0E27→28               LD    C,27H→28       ; C=27H→28      [ 4FH→50 ]
18 7018 EDB8                  LDDR
19 701A C9                    RET                  ; RETURN
20 701B                 ;
21 701B                       END
```

5月号　逆アセンブラの作成とその応用例

P.108 リスト3と4を以下のとおり訂正いたします。

リスト3　イニシャライズ

```
AF 11 00 00 CD 33 00 2A
71 11 22 0E 12 11 00 00
21 9E 8A B7 ED 52 4D 44
21 00 12 D3 E0 B0 C3 09
00             ED      00
```

（E0とB0の間にEDを挿入，09→00）

リスト4　オープニングメッセージ

```
30 20 20 ～
20 20 20 20 CF CF CF CF CF
CF CF CF CF CF CF CF CF CF
CF CF CF CF CF CF CF CF CF
CF CF CF CF CF CF CF CF CF
CF CF CF CF CF CF 20 0D 0D
00                0D 00
```

（20 0D 0D 00 → 0D 00）

○編集室から

8月号予告——7月18日(月)発売

●特集 MZ LOGOの魅力

●ビジネス・アプリケーション

ホームコンピュータ活用例の原稿，締め切り間近です

読者の皆さんの，ホームコンピュータとしてのマイコンの活用原稿を募集中です。MZ,X1,PCシリーズを，実際に家庭でまた職場で利用されていて，“これは便利，これは使える”というプログラムを公開してください。特に家庭事務管理，学習用，またはスモールビジネスの分野で役立っているプログラムを希望します。

有効な内容のもの数点を選出し，本誌掲載分には規定の原稿料をお支払いします。締め切りは6月末日です。ふるってご応募ください。

また，MZのクリーンコンピュータとしての特性を生かした，BASIC，マシン語以外のプログラム，活用例も募集します。小さいけれど，オリジナル言語を作ってみた，なんて方もぜひご投稿ください。せっかくのクリーン設計を生かさない手はない，という原稿をお待ちします。

送付先は，Oh！MZ編集室「ホームコンピュータ」，「クリーン活用法」係まで。

●本誌のお買い求めについて●

最近，本誌の品切れが相次ぎ，読者の皆様方にご迷惑をおかけしていることを深くお詫び致します。本誌は全国の大型書店，マイコンショップで販売しておりますので，予約ご購入が入手確実です。品切れのため，やむを得ず本社にご注文いただく場合は，送料実費をいただくことになります。

ご送金は郵便振替（東京1-29307）か現金書留で，ご注文書には住所，氏名と雑誌名何月号かを明記してください。なお，バックナンバー在庫等のお答い合わせは，出版部（03-261-4095）宛にお願い致します。

広告目次

Oh!MZ 7月号
◆1983年6月18日発行　定価480円
◆発行人　孫　正義
◆編集人　田鎖洋治郎
◆発売元　㈱日本ソフトバンク
出版部　〒102 東京都千代田区四番町2－1　☎03(261)4095 FAX 03(262)8397
本社　〒102 東京都千代田区九段南2－3－14　靖国九段南ビル☎03(263)3690㈹
TELEX 東京232－4614 JSBTY
大阪営業所　〒542 大阪市南区難波千日前5－19 河原センタービル3F
☎06(644)0191㈹　FAX 06(644)0160
◆印刷　凸版印刷株式会社

SHIFT BREAK

▶なんと今月は我家の引越し（もういいよ！）。さて家はともかく，重要なのは室内のレイアウト。non-no インテリアブックなどを参考に…，とはいっても，実際にはベッド，デスク，大小5つの書棚，オーディオ，ドラフターなどをいかにバランスよく配置するか？ MZで，Interior simulation ができるといいんだけど…。（T）

▶「新婚旅行なんだから休みをくれい」ともっともらしい理由をつけて北海道を旅行できることになった。なにせ，仙台より北は行ったことがないから未知との遭遇というわけで，まだ雪は降っているだろうかなどといらぬ心配。

ところで，入社以来1度もスカートをはいたことのない私のスカート姿を期待してる人!! 足は見えないよ。(K)

▶「今度こそ行こうよ」とOh！PC担当のG氏。「日光の奥がいい。今度の出張校正が終ったらね。金曜の夜に出発してさ。夜中のツーリングもいいよ。ぶっとばして疲れれば眠るの。ナポレオンがあるからね。星を探しているうちに意識を失うさ。沢の近くがいい。目が醒めるとあたりは緑でいっぱい。冷い水で顔を洗って…」 (ja)

▶今回のマイコンショウでは2種のLOGOが発表され，その1つは，新製品MZ-2200用のもので，米国UNISON WORLD社の開発。編集部では この，新世代のための言語を次号で特集する予定ですが，MZ-2000用も販売されるとのこと。開発担当者との独占インタビューも計画しております。乞御期待！ (M)

▶ビジネスショウに続いて，マイコンショウで変った製品は？ ポータブルコンピュータ，PC-5000に驚かれた人が多いと思う。学生用の英文タイプライターの大きさで，日本語ワープロも使えるポータブルコンピュータ，小型の本体の蓋を開けると，液晶ディスプレイが現われる現代版魔法の小箱!! (H)

▶本誌のK嬢が結婚する。この本が出るころにはめでたくゴールインしている予定だ。亭主を尻に敷くのか，貞淑な妻となるのか，野次馬どもの意見の分れるところだが，そんなことはどうでもよい。要は二人で協力しあって，独自の家庭を創造していくことだ。ところで尻敷きタイプか貞淑タイプか，やはり気になりますなあ。(O)

▶この欄を個人攻撃の場に悪用している，ということで次号から書かせてもらえないことになった。事実を語ることの困難，短い脚をなぜ長いと言わせようというのか。それでも地球は回ると言った哲人のなんと偉大なことか。歌を忘れたカナリア同様，沈黙しながら，それでも本誌は売れる！ ただそれだけが最後の願いです。 (T)

●編集　安田千尋　菊川良子　前田　徹
●技術　松田辰夫　小林初雄　徳永　聡
●海外協力　Hisayoshi Mikami　Ronald N. Billings　Hong Liang Lu　Ian Allen
●協力　有田隆也　秦野雅彦　青木秀浩　西畑文広　Itti Rittarporn
●カメラ　浜崎　昭　杉山和美
●イラスト　永沢しげる
●レイアウト　中島真子　岡谷勝人

日本ソフトバンク発行のOh!シリーズ

7月号
480円

Oh!PC以外にPCの専門誌なし

■パソコン誌で初の試み
"16万でできる"8インチ・ディスクドライブ

■内外のパソコンショウ特集

大型シミュレーションゲーム
■SPACE EMPIRE

好評連載
●16ビットの可能性！　●体験WordStar
●PC工作入門　●機械語熟語集

第3号
480円

HC-20をデッカク使おう！

第3号　好評発売中

■特集 データエントリマシンとしての試み
■新製品QC-10の全容
■システム内ルーチンの活用
■FP-80、RP-80試用記
■バーコードリーダの活用
■特別折込み付録バーコード表

第3号
480円

隔月刊になったOh!FM

第3号好評発売中

■マシン語特集
マシン語ってなんなのですか？
Oh!FM用統一チェックサム
マシン語プログラム実用例
演算のアルゴリズムとプログラム例／中間色PAINTもどきプログラム／サブシステムプログラム転送・実行など
■FM-8の命令を増やそう！
■三次元パッケージのアルゴリズム

Oh!
Oh!
Oh!
1983.5月12日誕生
満1才
満1才
PC
Oh!
MZ
55
Oh!
Oh!
生後8カ月
生後6カ月
HC
FM
1983.5.18

MDBA *Menu*

ART CREATOR

機能説明　　パソコンテレビ X-1 用　　3,800 円

コンピュータ・グラフィックスの世界で遊んでみませんか。ART CREATOR は、コンピュータで絵を描くためのソフトウェアです。
紙に絵を描くのと同じように、骨組みは太ふでを使って太く、大まかに描き、細かいところは細く、ゆっくり描けます。使えるコマンドは 15種類あります。色は単色の他に、カラーパターン自動生成機能により自由にカラーパターンをつくれます。また、ユーザー定義キャラクタをカラーパターンとして使えますので、3万色以上のカラーパターンを表現できます。

1. 四角、直線、円、楕円、多角形を簡単に描けます。
2. ぼかし、かすれ 等を描くブラシ機能があります。
3. ジョイスティックが使えます。
4. 640、320両モードで描画できます。
5. コーザ定義キャラクタを定義できます。
6. PALET 変更、カセットへの記録 他、機能を満載しています。

SIMW16003

4人ポーカー

MZ-2000, MZ-80B, X-1,　　2,800円

機能説明

1. あなたと3台のコンピュータとの対戦です。持ち点は 300 点、誰かがゲームに参加できなくなると終りです。
2. 5枚のカードが4人に配られ、先頭の人からカードを交換します。あなたがいつも先頭ではありません。勝った人が先頭なのです。人間とコンピュータとは対等に扱われます。
3. 交換が終ったら、点をかけてください。かけひきは巧妙です。ドロップすると、人間不在のままコンピュータ同士で勝手にゲームを進めます。

あなたは勝つ前にどれだけゲームに参加できるかな？

FUMW13001
SHMW18002　SIMW16001

エコノミーパック PART-1

	A	B	C	D
コウモクメイ				
1				
2				
3				

MZ-2000, MZ-80B用　　4,800円

機能説明

住所録とノンプログラミング言語の汎用集計表がパックになった、お買い得な ビジネスソフトウェアパッケージ です。

1. 240名分の〒、住所、氏名、TEL、メモを記録することができ、かつ、10の区分を設定することができます。
2. 東京にいる佐藤さんで、高校の友人だった人のラベルを出力、大阪の人だけ全部出力など、各項目を自由に指定できる多重検索機能がついています。
3. 汎用集計表では、算式登録が自由に行なえ、一度算式と項目名を登録すれば、あとは異なるデータで何度でも利用できます。
4. 26×80のデータの演算、並べ替え、プリンタからの出力、カセットからの LOAD、SAVE ができます。

SHMW18003

MUSIC EDITOR

MZ-2000, MZ-80B用　　2,800円

機能説明

1. オルガンをひく感覚で MUSIC 文を作成します。脇ってひいても一音ずつの削除、一文ずつの削除ができます。
2. 25KB (25,000文字) 分の音を記録し、連続出力します。カセットに SAVE、LOAD が行なえ、LOAD 時にはアペンド (追加) 機能があり、フレーズ単位の音楽の編集ができます。
3. プリンタから MUSIC 文を出力できます。
4. 使い易いワンキー・オクターブ変更機能。
5. デモン ストレーションファイル (UFO) がついています。
6. USE R'S MANUAL (使い方説明書) がついています。

SHMW18001

SKY DIVING

機能説明　　MZ-2000, MZ-80B 用　　2,800 円

君は名高い冒険野郎だ。きょうのチャレンジは、悪天候の中を、海面に浮かぶリモート船をコントロールしながら降り立つことにある。降下中、カモメにパラシュートを破られないよう、また、海には巨大なサメがいるので、船を破壊されないように注意されたい。落雷が時々あるそうである。では、幸運を祈る。

1. 降下開始はスペースキー。
2. パラシュートを開くのは TAB キー。
3. ←→ キーによりパラシュートをコントロールする。
4. カモメをけるのは、スペースキー。
5. ↑↓ キーにより船をコントロールする。

SHMW18004

BATTLE ARMY

パソコン テレビ X-1用　　2,800円

機能説明

時代は、人間とコンピュータとの権利平等の時代から、コンピュータ絶対王制の時代へと移りつつある。我々は、人類の権利を死守すべく、君に BATTLE ARMY 軍の全指揮を委ねた。君の任務は全軍を率い、敵総指令部を占領することにある。
敵軍が全てコンピュータのコントロール下にあるからといって、怖れず、全力を尽くし、勝利の女神を我々人類に傾らしたまえ。
(ルールの詳細は、プログラム開始後説明します。)

SIMW16002

マイクロ データ
ベース アソシエイツ
☎(0427)35-1600

SUPER SOFT WARE
from I企画

MZ-3500に本格的ワープロ・ソフト新登場！

文筆 MZ-3500用
5インチ両面（MZ-1004） ￥48,000（D）
※システムディスクの他に熟語・短文用ディスク付。

システム構成

本　体：MZ-3500(漢字ROM付) 2ドライブ以上実装。G-RAMの有無は問いません。
ＣＲＴ：MZ-ID02(400×640・グリーン)、MZ-ID03(400×640・カラー)
プリンタ：MZ-IP03(24×24ドット)
MZ-IP06(16×16ドット)
※プリンタごとに別製品になっています。
指定構成機器以外での使用においては、ソフトの責任を負いかねます。

機　能

入力方式……カナ漢字変換入力・16進コード入力
編集機能……センターリング、右寄せ、n文字挿入・削除、一行挿入・削除。
熟語登録……3文字まで、最大24,000語。
文書作成中に登録可、熟語リストのプリンタ出力可。
熟語呼出時の自動学習機能有。
短文登録……100文字まで、100文。
文書作成時に登録可、短文リストのプリンタ出力可。
画面出力……40桁×16行、ROLL・UP、ROLL・DOWN、縮小画面（文書作成中に可能）
印　刷……1ページに40文字×39行まで、連続印刷可、行間隔・文字間隔指定可、拡大文字の混在可、自動縦書機能。
文書保存……72頁（DISK 1枚当り）
その他……アンダーライン（MZ-IP03は不可）

★詳しい資料ご希望の方、下記宛までご請求下さい。

ソフトハウス I企画 株式会社 アイ企画
〒530 大阪市北区同心2丁目5番16号
岡本ビル ☎06-352-7137
東京
〒170 東京都豊島区南大塚2-45-11
メゾンドールアム ☎03-943-7451

5"デュアルディスクユニット

快走!disk-80markII

大好評のdisk-80markIIシリーズ。高速に加え、さらに静寂・高性能。独走中。

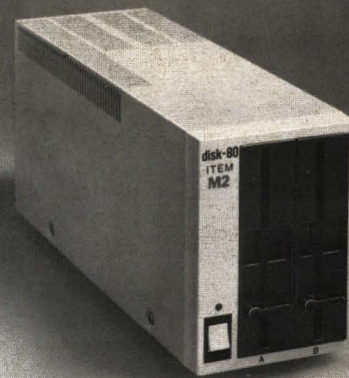
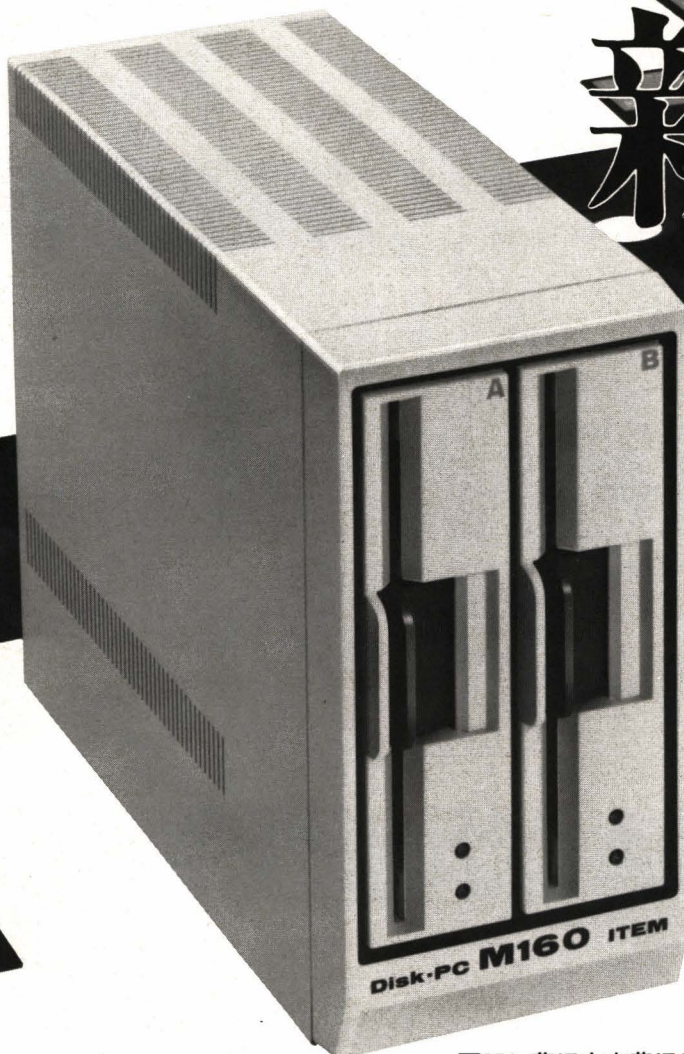
PC8801/PC9801/PC8001markII/PC8001専用
disk-80PII　123,000円

■RACET NEC DOS、CP/M対応 ■PC8001でご利用の場合はdisk-I/O P(12,000円)と接続ケーブルCBL-1(5,000円)をご利用下さい。他はCBL-2(5,000円)でダイレクト接続。システムディスケットは各対応のものをご利用下さい。

FM-7/FM-8専用
disk-80FII　128,000円

■DISK BASICをはじめFLEXなど各DOSに対応。
■FM-7の高速バージョンの対応も可能です。■接続にはdisk-I/O F(15,000円)とCBL-3(5,000円)をご利用下さい。

MZ-80B/MZ-2000専用
disk-80BII　138,000円

■フロッピーインターフェイス内蔵。拡張I/Oポートがあればあとは接続ケーブル(CBL-3)でダイレクトIN。

SOFT新時代

高性能パワフル8インチフロッピーディスクユニット

Disk-PC M160

新発売 標準タイプ **256,000円**

CP/M、MS DOS、P-SYSTEM…………
数々のパワフルなSOFTが続々登場してきます。まさにSOFT新時代。

スリムな薄型設計。大容量1MB×2ドライブ。
ビジネスに、パーソナルデータベースに、
ハイコストパァーフォーマンスを提供いたします。

モデル

M160-01(PC9801用)…………………256,000円
M160-02(PC9801用CP/M-86付)……298,000円
M160-03(PC8801用)…………………296,000円

※M160-02以外はシステムディスク(DISK BASIC等)が必要です。接続ケーブル、インターフェイス等はふくまれています。

□お求めは全国マイコンショップへ。
□電話によるご注文もうけたまわっております。
0466-27-1668(AM9:00~PM8:00)
□詳しいお問合せは左記あてへお申込み下さい。
□disk専用クレジットが用意されています。

株式会社 アイテム
〒251 藤沢市南藤沢8-1-202
TEL (0466)27-1668(代)

*CP/M®はデジタル・リサーチ社の登録商標です。*UCSD Pascal ™はカリフォルニア大学理事会の登録商標です。*FLEX®はTSC社の登録商標です。

連続ゲーム

特報 日本初の本格的ロールプレーンゲーム

徳川家康

(X1、MZ-700、MZ-80B、MZ-2000、FM7)

あなたはこの連続ゲームに勝ち関ヶ原を手中に!!

人の一生は重荷を負うて遠き道をゆくが如し 急ぐべからず 家康

● 竹千代(幼名)の人質時代の苦難を逆擬人化の手法によりロールプレイングゲームとして登場!!

STORY

① 少年編 (¥4,000)

人質にとられることになった竹千代は、岡崎城をあとに駿府へ旅立とうとしていた。その時すぐ目の前が煙につつまれたかと思うと、巨大なヘビが姿を現した。竹千代は突然の出来事に立ちすくんだ。はっと我に返って見ると、そこにはもう自分と2人の家来しかいなかった。逃げてもむだと子供ながらもさとった竹千代は、2人の家来と共に戦った。戦いの後に待っているのは何か!?脳裏に不安がよぎる。振り返ると城は門をとざし、しんと静まりかえっており、門の開く気配もない。3人はしかたなく町々で仲間を集めながら駿府へとむかった。天文16年9月のある日のことであった。

● 家康の人生の転期となる、丸根の砦をめぐる戦い!!

STORY

② 桶狭間決戦前夜 (¥4,000)

ある風もない静かな夜、家康はいきりたつ家臣をよそに、ひとり悩んでいた。あすの戦いのことである。初陣は大成功に終ったが、やはり不安であった。それもそのはず、敵はあの織田信長なのだ。家康自身にとっては、なんら信長と戦う理由はない。しかし……家康はまだ人質のような身の上。戦わざるを得ない運命にあった。

眠れない夜を過しつつ、ついに戦いの朝を迎えた。戦いの用意も整い、家康一隊は、丸根の砦を気にしつつ布陣を始めた。まさに時は今、一本の鏑矢が風を切り、静寂を破った。血湧き!肉躍る戦いの始まりである……

NEW漢字ワードプロセッサー

MZ-700のユーザーに解放!!

プログラム多数あり!

● 特長
- ★漢字 ROM もフロッピーもいらない最低コスト
- ★1度使った文章をカセットテープに保存
- ★文章の追加、削除、訂正が簡単
- ★漢字は登録番号を入力、ひらがな・カタカナ・アルファベット等はKEYから入力

漢字

```
1:  615,1317,セズ",3,オ,
2:  &,パ゜ソコン,&,ト ,&,プ゜リンター,&,ダ゛ケデ゛,
3:  615,エル,&,テープ゜ベ゛ース,&,ノ,/
4:  575,1006,&,コアト,&,ノ ,319,649,&,
5:  ワードプ゜ロセッサー,&,デ゛ス。,/
```

漢字ROMも フロッピ゜ィデ゛ィスクも使用せす゛、お手持ちのパ゜ソコンと プ゜リンターだ゛けて゛使えるテープ゜ベ゛ースの最低コストの 漢字ワード゛プ゜ロセッサーて゛す。

●テープの複製不可 ●漢字は画面に表せません。

面白さ抜群!! X-1ソフト

●花札(オイチョカブ)ほか ●7 STUDE PORKER ●TV将棋盤 ●四人麻雀 続々登場

※上記の他にも多数のソフトを開発製作しています。 最寄りのマイコン店または当社までお問合せ下さい。

ウスヰパソコンセンター
〒930 富山市総曲輪3-6-3 ☎(0764)21-4181

MZディスクユーティリティー

●MZ-1200, K/C
●MZ-80B
●MZ-2000
各9,800円(〒200)

※MZのディスクを使いこなすために，DUMP, DATA, MAP, COPY, XFER, MASTER, 等々10数種のコマンドを用意しました。〔シングルフロッピー，プリンター出力可〕

★詳細は、Oh!MZ2月号参照。SOFT BANK加盟店でお求めいただけます。

例えば…

- ●スタート・エンドアドレスのわからないプログラムのバックアップ。
- ●テープのシーケンシャルデータをディスクへ転送
- ●ディスケット間で、必要なプログラム・データのみの転送
- ●マスターディスケットのバックアップ。
- ●ディスケットの内容をセクター単位で読み出し，書き換え。
- ●オリジナルマスターディスケットの作成
- ●うっかりDELETE一発復活法(K/C以外)
- ●F-DOS, CP/Mを使用中にうっかり消してしまったファイルの復活(B, 2000用のみ)
- ●B用は，Hu-G BASICのディスクでの起動法有

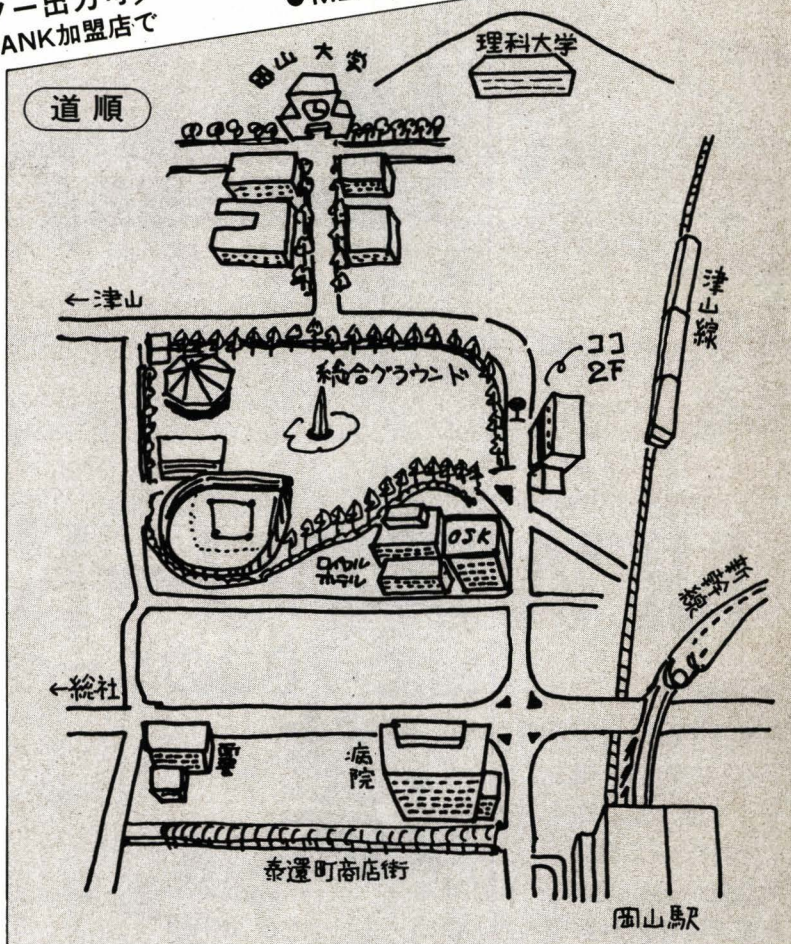
などなど……とにかく便利!!

オリジナルNewSOFT

※Super Machine Language

Super Machine Lagguageは、マシン語入門の方に、そしてシステム開発のソフトとして、シャープ社のマシンランゲージSP-2001を更に使いやすく、機能的なマシン語モニターとして作られました。スクリーンエディターの強化に加え、メモリーダンプのスピードアップ・命令単位でのメモリーダンプ・指定アドレス間のチェックサムの計算・プログラムリロケート・オペコードサーチ・アスキーコードサーチ・メモリーペイント・テープコピー・コールコマンド・オープニングメッセージ画面の作成……等便利なコマンドも豊富に追加!

また雑誌などのプログラムダンプリストの打ち込みもスピーディーに，かつ間違いなく入力できる様、入力時に8バイトチェックサムが表示され入力ミスの発見も大変楽になります。“SP-2001”のセカンドモニターとしても十分価値ある1本!!

●MZ-80K/C, 1200用
●MZ-80B, 2000用 各3,800(〒200)

岡大・理大生……岡大正門通り53号線交差点左折　岡山駅方向へ徒歩5分左手側

岡工生…………スポーツ公園横断、正面入口向側

バスご利用の方…駅前3番ホーム14番のりば 津高営業所行 スポーツセンター前下車(10分間隔)

WORLD WIDE COMPUTER SUPER SHOP

COSMOS™ 岡山

〒700 岡山市南方5-6-5 今田ビル2F（県営グランド入口向）
TEL(0862)54−7474　ミニファクス(0862)54−7481
年中無休　AM10:00〜PM7:00
通販は現金書留又は郵便振替 岡山4-12524 コスモス岡山

MZは不滅だ!!全国MZ10万人ユーザーに贈る最新ゲーム

MZ-700：1200K/C用 GAME SOFT

COSMOS明石 MARVEL SOFT WARE

●ソフト送料／1本¥200／5本以上無料

PCGスペースクライマーはMZ700用とMZ1200K／C用の２種類あります

PCGスペースクライマー　¥3,500
MZの限界に挑戦した最新ゲーム、とにかく最高です。

PCGモンスターパニック　¥3,500
MZの限界に挑戦した最新ゲーム、ゲームをしながら音がなる。

PCGスーパーガブラー　¥3,000
エイリアン、MZ、ハンマー、ノコギリetc.回転ドアを利用してエイリアンをやっつけろ。

PCGスーパーモンスター#2　¥3,500
高速タテ、ヨコスクロールが最高　モンスターをブロックではさみ打ち

PCGスペースCITY　¥3,000
宇宙都市を救えだんだん、レベルが高くなり、だんだんむつかしくなる。

PCGスーパーパックル　¥3,000
キャラクターのおもしろさ最高！君はおいかけ、モンスターからエスケープできるか！

PCGプロレスモンスター　¥3,500
プロレスファンに贈る最新ゲーム　MZは今プロレスのワンダーランド。

PCGアステロイドパニック　¥3,000
舞いおりるエイリアンをやっつけろ、キャラクターのおもしろさ最高！

PCGスペースUFO　¥3,000
ゲームセンターの様な本物そっくりの楽しさ最高のゲーム！

PCGパルサーII　¥3,000
そこは、エイリアンの巣窟だった。あなたは一画面消す事ができるか。

スネークワールド　¥2,500
ライフルでヘビとサソリをやっつけろ！

PCGスネークワールド　¥3,000
ライフルでヘビとサソリをやっつけろ。

5カードポーカー　¥2,500
H／Lゲームでおもしろさ倍増、コンピュータと一騎撃ち。

迷路脱出ゲーム　¥2,500
時間内に迷路から脱出するゲーム。

アステロイドパニック　¥2,500
舞いおりるエイリアンをやっつけろ。

神経衰弱ゲーム　¥2,500
２人で楽しめるゲームです。

MZ700＋グラフィックボード＋PCGゲームソフト3本(MZ721)¥110,000
グラフィックボード取り付けしてお送ります。送料は当社負担。

当社オリジナルゲームは、HAL PCG-8000ロータスゲームボードが必要です。

PCG使用ソフトは、本体内にグラフィックボード取り付けが必要です。C/K、K2、K2E、1200、700ユーザーはグラフィックボード取り付け可能です。

※ボード取り付けの注意
CPUを基板から取りはずし、工作が必要です。加工に自信のない方は、本体をお送り下さい。取り付けいたします。

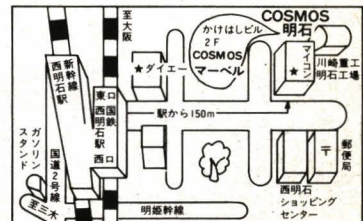
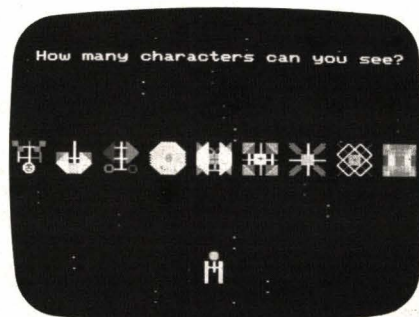
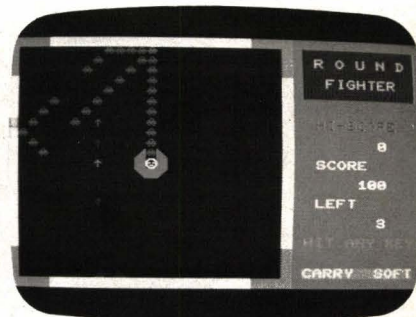
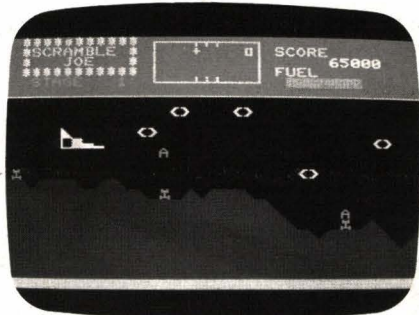
グラフィックボード取り付け(送料コミ)…… ¥30,000
グラフィックボード(ユーザーで取り付けの場合)…¥20,000
《機種名記入でお送り下さい。》

オリジナルソフト買います。
詳しくは COSMOS明石
企画室 吉田まで
078(923)5536

全国通信販売のご案内

お求めは全国のソフトバンク加盟店まで直接当社に御注文の場合、商品名を明記の上、現金書留でお願いいたします。

COSMOS明石
(株)マーベルコンピュータ
〒673 兵庫県明石市西明石南町1-10-13
☎078-923-5536 ミニファクス
●定休日 毎週金曜日　●営業時間／AM10：00〜PM8：00

CARRY LAB

ソフトも ターボ感覚。

スターブラスト MZ-700用
マシン語／CG-92／¥3,000
飛びまわるエイリアン、すべてのエイリアンをやっつけろ!!

ラウンドファイター MZ-700用
マシン語／CG-93／¥3,000
編体を組んでせまる敵をつぎつぎにやっつけろ!!

スクランブルジョー MZ-700用
マシン語／CG-94／¥3,000
この宇宙をどこまで行けるか!もう宇宙は燃えている!!

レーダースネーキー
PC-8001用
PC-8001 Mk.II N-BASICモード用
PC-8801 N-BASICモード用
マシン語／CG-95／¥3,000
逃げまわるカエルを食べろ! でも、食べれば食べる分だけ太ってしまいます。ご用心!!

◎新しいパートナー
マイコンハウスSPS
マイクロキャビン

(有)キャリーラボ
熊本市大江6丁目25-25 金子ビル1F 〒862
TEL 0963(63)0211
電話ファクシミリ 0963(63)0235・G2・G3

主な販売店 富士音響 (株)ニデコ (株)近畿システムサービス 上新電気 ダイイチ産業　通信販売 御注文は現金書留か郵便振替を御利用ください。尚、送料として300円加算してください。郵便振替口座 熊本18846

MZシリーズ

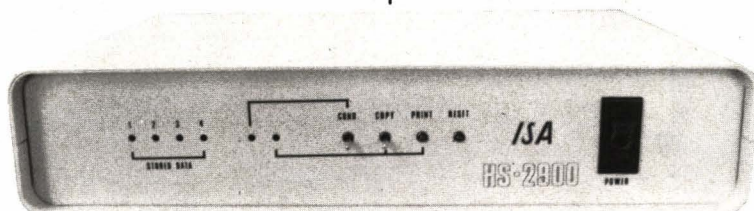
シャープMZシリーズにバッファ
が接続できるようになりました。
また従来の機種も接続できます。

コンピューターシステムの
処理能力が増強されます。

高速バッファ装置 **HS-2900**

HS-2900(セントロニクス仕様)…¥68,000
HS-2901(RS-232C仕様)………¥90,000
HS-2902(GP-IB仕様)…………¥110,000

- 高速バッファリングが可能
- 大容量バッファリングメモリ 62K Byte(約6万文字)
- リピートモードにより同一コピーが可能
- 2つのモードを持ったセルフテスト機能
- コンデンスモードによる空白等の凝縮が可能（12万文字以上も可能）
- バッファ残量表示によりバッファの状態が監視可能
- 複数台のシリーズ接続が可能
- 高信頼性設計

■セントロニクス・パラレル
■RS-232C
■GP-1B

■セントロニクス・パラレル
■RS-232C

プロッター

プリンター

ASRタイプ
コンソール

テープ
パンチャー

シャープMZシリーズ、X-1に接続可能なプリンター

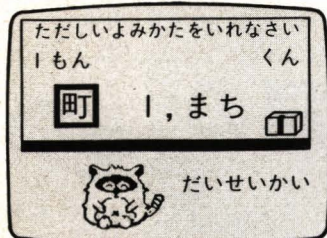
●MZ-80P4B ●MZ-80BP5 ●MZ-80P6 ●MZ-1P04 ●MZ-80P4K
●MZ-80KP5 ●MZ-1P02 ●MZ-1P03 ●CE-333P ●CE-330P
●CZ-800P ●SEIKOSHAプリンター ●EPSONプリンター ●その他

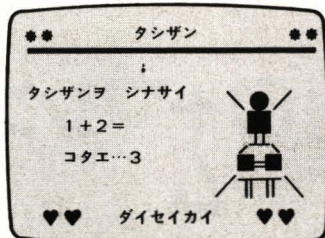
STRATFORD SOFT

監修 家庭教師センター 学習館

STRATFORD C.A.I.

全種マニュアル付

楽しい国語〔漢字〕
小1・小2 要V-RAM1 ￥3,800

MZ-2000・80B

楽しい算数
小1～小5 各上巻発売中 ￥3,300

MZシリーズ, PC-8001・6001

世界地理〔河川・海流編〕
要V-RAM1 ￥3,800

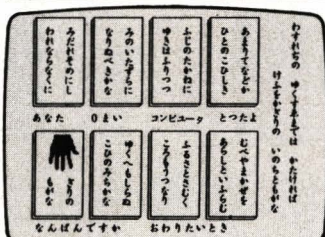
MZ-2000・80B

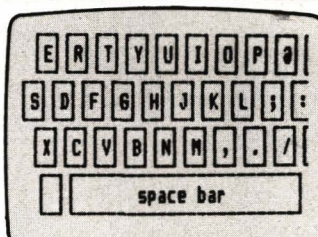
クラス別成績管理
教職員向 ￥4,500

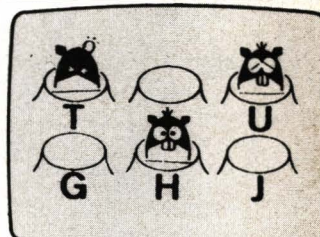
MZ-2000・80B, PC-8001, FM-7・8

速習英文タイプ
￥3,000

MZ-2000・80B, PC-8001, FM-7・8

ひらがな百人一首
要V-RAM1 ￥4,500

MZ-2000・80B

キーボードトレーナー
￥3,000

MZシリーズ PC-8001, FM-7・8

モグラたたきキーボード練習
近日発売 ￥3,000

X1, JR-200

パソコンテレビX1用 日本語ワードプロセッサ、近日発売！

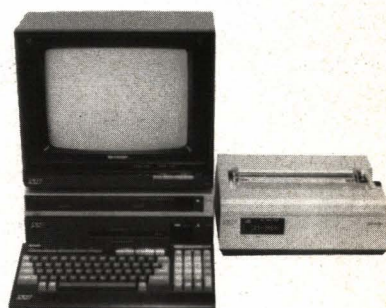
ビジネス事業部

あなたのX1がワープロに！ホームユースからビジネスユースまで、どなたでもその日から手軽に扱えます。

(実際の画面は37文字×10行です)

読みからの漢字検索　JISコードからの漢字を検索／熟語検索／漢字登録／熟語登録／文章登録／文章検索／文字印刷／文章編集

原稿用紙 約400枚分の文章を一枚のディスクに保存

〔要漢字ROM〕 予価38,000円
5インチフロッピーディスク版

★レベルⅢの教育用ソフトは総てブック型パッケージ入りです。価格は、各3,800円です。

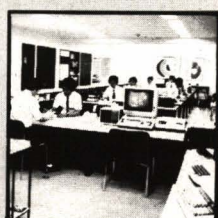
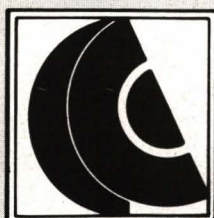
★購入を希望なさる場合は上記のプログラム名(教育ソフトの場合は教科書名や学年、上下別も)・機種・電話番号を明記の上、現金書留で当社Oh! MZ 係までご注文下さい。〔郵送料は不要です。〕

★当社ソフトウェア総合カタログをご希望なさる場合は、住所・氏名・年令・職業を明記の上、資料請求券と切手200円分を同封して当社までご請求下さい。

ストラットフォード・コンピューターセンター株式会社

STRATFORD COMPUTER CENTER CORPORATION

〒336 埼玉県浦和市南浦和2-36-15 ［南浦和東口駅前・埼銀隣り］ ☎0488(85)5222大代表

全　景

ソフト開発部

シンボルマーク

編　集　部

授業風景

資料請求券 Oh!MZ⑦

ユニークなオリジナルソフトで躍進する

CHAMPION SOFT

好評発売中ソフト

(FM-7)

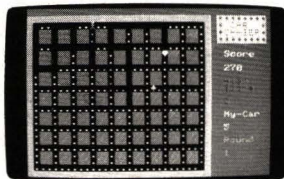
君の人生は平穏無事か。それとも波らん万丈か。
君の人生をシミュレートしてみよう。

●作／白木善喜　各¥3,500

BASIC+マシン語の高速ゲーム　新発売

カーチェイサー／各¥3,500

(MZ-700)

適合機種：
MZ-2000・80B
MZ-700,
PC-8001MKII
・8001, JR-200
パソコンテレビ
X-1, FM-7

コンピュータがコントロールする車の追跡をうまかわしながら、路上のドットを喰い逃げできるか……。

●原作／谷田宣文●移植／籔内雄美　各¥ 3,500

ジャンピングクロス／各¥3,500

(MZ-700)

適合機種：
MZ-2000・80B,
MZ-700,
PC-8001MKII
・8001, JR-200
パソコンテレビ
X-1, FM-7

Jumperを操り、右端から流れてくるMeteor(あるいはUFO)を避けながら蔓に下っているダイヤモンドをいくつとれるか……。

●原作／谷田宣文●移植／籔内雄美　各¥ 3,500

シミュレーションビジネスウォーゲーム

コンピュータウォーズ／各¥3,500

(PC-8001MKII・PC-8001)

適合機種：
PC-8001MKII・
PC-8001, FM-7
MZ-2000・80B,
MZ-700
パソコンテレビ
X-1

君はコンピュータ会社の営業部長。現在10%のシエアを3年後に50%にできるか……。

●作／白木善喜　各¥ 3,500

シミュレーションウォーゲーム

コンバット-47／各¥3,500

(FM-7)

適合機種：
PC-8001MKII・
PC-8001, FM-7
MZ-2000・80B,
MZ-700
パソコンテレビ
X-1

1～4人で楽しめる知的戦略ゲーム
47か国のうち君は何か国を占領できるか。

●作／白木善喜　各¥ 3,500

コンピュータダイスプレイゲーム

ウエディングレース／各¥3,500

(パソコンテレビX-1)

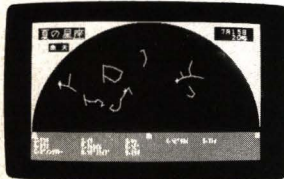
適合機種：
PC-8001MKII・
PC-8001, FM-7
MZ-2000・80B,
MZ-700
パソコンテレビ
X-1

君はめでたく結婚にゴールインできるか。それとも破産のうきめにあい、地獄行となるか……。

●作／白木善喜　各¥ 3,500

星空の旅

ザ・プラネタリウム／各¥3,500

(パソコンテレビX-1)

適合機種：
PC-8801,
PC-8001MKII
MZ-2000, FM-7
パソコンテレビ
X-1

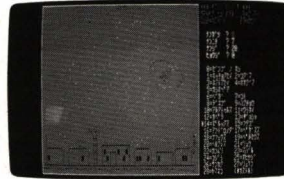
北天・南天別に四季の星座を満喫できます。

●作／白木善喜　各¥ 3,500

楽しい星座教室(マニュアル付)

ザ・スターズ／各¥3,800

(FM-7)

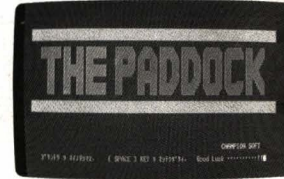
適合機種：
PC-8801
PC-8001MKII,
MZ-2000・80B,
FM-7
パソコンテレビ
X-1

北天・南天別に月，日，時刻を変えてインプットすると各星座の動きを見れます。
また、星座を拡大して見ることもできます。

●原作／館盛光弘●移植／北村隆次　各¥ 3,800

科学的競馬予想(マニュアル付)

ザ・パドック／各¥3,800

(パソコンテレビX-1)

適合機種：
PC-8001MKII・
PC-8001, FM-7
MZ-2000・80B,
FM-7・8
パソコンテレビ
X-1 JR-200

①持ちタイム②前走タイム③脚質～等14種のデータを入力して、勝馬を予想し連複券を推せんします。

●作／ビックスマイコ　各¥ 3,800

発売元 エヌ・テイ・エル産業株式会社
〒657 神戸市灘区篠原北町1丁目5番13号
TEL (078) 881−1914
ファックス (078) 882−0555

制作 株式会社チャンピオンソフト
〒530 大阪市北区西天満5丁目15番18号(実業ビル)
TEL (06) 365−9900

ユニークなオリジナルソフトで躍進する

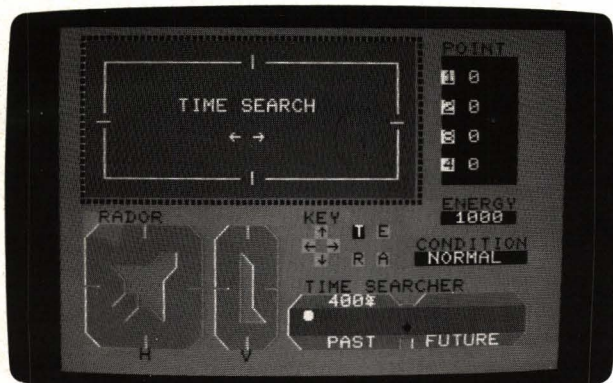
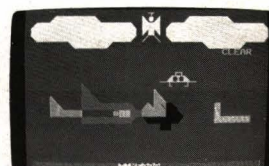
CS CHAMPION SOFT

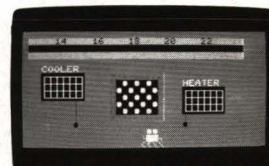
自信の新作ソフト続々登場

(新登場) NewGame

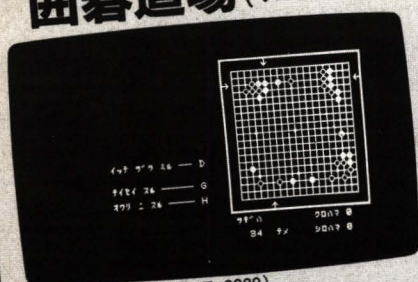
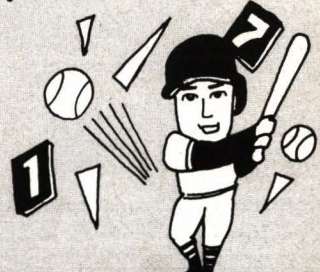
(写真はMZ-700)

タイムパトロール

君はゲームの中のゲームを見たか？
ナ・ナント！ゲームの中にゲームが4つもあるのだ
宇宙のパトロール隊員になるためには、4つのゲームすべてにチャレンジしなくてはならない

●オリンピックハイジャンプゲーム

●人類誕生ゲーム

●遺伝子組替えゲーム

●お天気ゲーム

■適合機種：MZ-700, PC-8001MKⅡ・8001, FM-7, MZ-2200, パソコンテレビX-1

■原作／平岩徳一■移植／夏目明彦 各¥3,800

(株)チャンピオンソフト
事務所移転のお知らせ

業務の拡大に伴い、6月25日より下記に事務所を移転します。
新事務所住所／〒530 大阪市北区西天満6丁目1番12号 瑞穂ビル
(電話番号は従来通りです)

発売元 NTL エヌ・テイ・エル産業株式会社
〒657 神戸市灘区篠原北町1丁目5番13号
TEL (078) 881－1914
ファックス(078) 882－0555

制 作
〒530 大阪市北区西天満5丁目15番18号(実業ビル)
TEL (06) 365－9900

ツクモ全店マイコン買取り、下取りグレードアップセール!!

お勧め入門機組み合わせセット

シャープ MZ-700組み合わせシリーズ

CMT-140R
標準価格￥67,800

サンヨー製14型中解像度RGBカラーモニターTV くっきり鮮明な画像を再現します。

MZ-731 ￥128,000
MZ-721 ￥89,800(プリンター別売タイプ)

ツクモオリジナル
ジョイスティック
JOY-700 ￥5,800

Aセット MZ-721(￥89,800)＋JOY-700(￥5,800)＋ツクモサービスソフト10本(￥25,000)＝￥120,600➡￥89,800

Bセット MZ-721＋三洋 CMT-140R(￥67,800 接続ケーブル￥1,800)＋JOY-700＋ツクモサービスソフト5本(￥12,500)＝￥177,700➡￥128,000

Cセット MZ-721＋ニデコム NH-12D9(￥99,800 接続ケーブル付)＋JOY-700＋ツクモサービスソフト5本＝￥207,900➡￥162,000

各セット送料￥2,000

ツクモサービスソフト5本
￥12,500

MZ-731(￥128,000)での組み合わせセットを御希望の方は、上記の各セット価格に￥32,000を加算した額が各セット価格になります

TVも気になる‥マイコンも気になる‥欲張り派の君へ

日立 C14-452 ￥79,800

今月のスペシャルセット

日立 14インチ2WAYモニターTV＋MZ-700組み合わせセット

主な仕様
- ○TVとモニターの切換えは前面でワンタッチ
- ○RGB入力端子付
- ○画面位置調節ツマミ付
- ○40×25文字(1000文字)表示可能
- ○スモークドフィルター付
- ○モニターTVとして使用時でも、TVの音声のみはモニター出来ます。

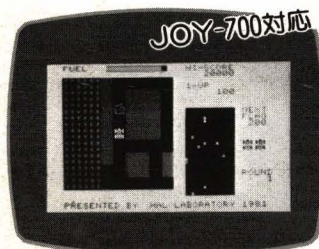
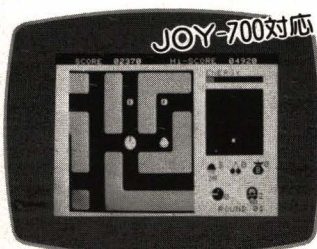
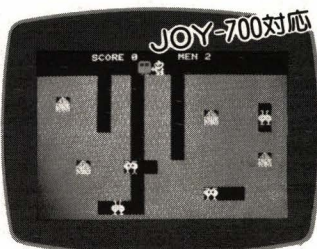
欲張りセット

MZ-721＋C14-452(日立モニターTV￥79,800 接続ケーブル￥1,800)＋JOY-700＋ツクモサービスソフト10本＝￥202,200➡￥135,000送料￥2,000

負けるな!!MZ-700「グラフィック機能」＋「サウンド機能」

HAL研究所製 PCG-700
￥29,800

MZ-700ファン待望のメカ「PCG-700」￥29,800で遂に登場!!

- ●PCG-700は"Programable Character Generator"の略称でMZ-700が持っているキャラクターを自由に作り変える事が出来る画期的な周辺機器です。
- ●最大256種類のキャラクターが定義出来ます。
- ●再定義により、無数の文字が表現可能です。
- ●MZ-700のメモリーは使用しません。
- ●PCGのON/OFFは基板上のスイッチ又は、プログラムで行ないます。
- ●PCG-700は、バスタイプですのでCPUを取りはずす必要はありません。

スペースラリーZ ￥3,000送料￥300

メイズパックマン ￥3,000送料￥300

ディグダイバー ￥3,000送料￥300

その他

○パックマン(JOY-700対応)
￥3,000送料￥300

今、入手のチャンス!! **PCG-700組み合わせシリーズ** PCG-700＋ツクモサービスソフト3本＝￥38,800➡￥29,800送料￥1,200

特報 PCG-8800(HAL研究所製高分解グラフィック装置￥44,800)＋ツクモサービスソフト5本(￥17,500)＝定価￥62,300➡￥45,800送料共

御注文の際は、御注文になりたい商品名を御記入の上、現金書留にてお願い致します。20万円以上は銀行振込みも取り扱っております。お問合せ下さい。

富士銀行　神田支店
普通口座　№894047

即決クレジット ツクモ全国クレジットOK!!
- ●現金特別価格で各種クレジットが利用できます。残金のみに金利がかかります。
- ●60回払いまで出来ます。但、1回の支払い額は3,000円以上。
- ●その場で、お持ち帰りできるクレジットもあります。
- ●印鑑、身分証明書(免許証など)が必要です。尚、学生の方及び、未成年の方は、ご両親の保証も必要です。
- ●取扱いクレジットカード:日本信販、JCB、DC、UC、VISA、MC、CF

■通信販売は 〒101 東京都千代田区神田郵便局私書箱135 九十九電機(株) MZ7係

九十九(ツクモ)電機 株式会社

店舗	〒	住所	電話
ニューセンター店	〒101	東京都千代田区外神田1-16-10	☎03(251)0986～8
名古屋店 3F	〒460	名古屋市中区大須3-30-86 ラジオセンター・アメ横ビル内	☎052(263)1681
5号店	〒101	東京都千代田区外神田3-1-14	☎03(251)0531～2
7号店	〒101	東京都千代田区外神田1-9-9	☎03(253)4199
札幌店	〒060	札幌市中央区南二条西3-15-1 さっしんビルB1	☎011(241)2299

■定休日 東京各店は毎週木曜日と第3水曜日、名古屋店は毎週月曜日

通信販売ご利用の方は、電話番号、年令も必ずご記入下さい。

一体、どこまで進歩するのか。Telly-BASIC

続々登場！

MZファンなら見逃せない。
あのMZ-1200/K/Cのディスク容量が、大きく2倍にアップ。

絶賛発売中！

MZ-1200/K/C用
倍密度フロッピー・インターフェース・カード　FDC-10H
標準価格　35,000円

●MZ-80K/K2/K2E/C/1200のディスク装置が倍密度ディスク(ディスケット容量が280Kバイト)に変わります。※シャープ製は単密度で、容量はこの半分です。●従来のフロッピーディスク装置やケーブルは、そのまま使用可能。しかし、マスターディスク(SP-6010など)は使えません。当社製のBAS-10Dをご利用ください。●このカードには、4KバイトのPROMが内蔵(含むモニター)されています。※PROMの主なサブルーチンは、マニュアルに公開済み。●メモリアドレスF000 ～FFFFH、I/OポートF8～FDHを使用しています。※このカードを使用するには、RAM48Kバイトが必要です。

好評発売中！

MZ-1200/K/C用
Telly-BASIC(倍密度ディスク版・10進10桁演算)　BAS-10D
標準価格　10,000円

●FDC-10Hと、共に使用します。●シャープSP-6010とテキスト・コンパチブルです。※機械語を使うものには、働かない場合があります。
●シャープSP-6010と比べ、次の命令が機能がアップしました。

■プログラム編集,デバッグ機能
▶APPEND：メモリ中のプログラムにカセット、またはディスク内のプログラムを付加します。これにより、汎用サブルーチンのパッケージ化が可能になります。▶DELETE：プログラム中の指定行を削除します。▶LIST/P：LISTをプリンタに出力する際、ページごとに指定したタイトルを表示。さらに、ページの上下に余白をとり、プリントアウト可能な機能を追加。従来に比べ、リストが大変見やすくなりました。▶SEARCH：プログラム・テキスト中の任意の文字列を検索し、画面、またはプリンタに出力します。▶TRON,TROFF：トレースモードをON/OFFします。ONにした場合、実行された行番号を画面、またはプリンタに表示します。▶VLIST：プログラム・テキスト中に使用されている変数の一覧を、その変数が使われている行番号と共に、表示します。▶KEY,KLIST：Telly-BASICでは、すべてのキーをユーザーが定義することができます。例えば、キーボードの右側をテンキーとして定義すれば、テンキーとして。また、それらのキーに、GOTO、GOSUB…・などのコマンドおよびステートメントを定義すれば、BASICの命令を1回のキー操作で入力することが可能です。

■ステートメント
▶DIM:1次元配列の添字の制限がなくなりました。(メモリの残量により制限されます。)▶HEX$:数値を、16進数を表わすストリングに変換します。▶INSTR:ストリング中から、他のストリングを探します。▶SWAP:2つの変数の値を変換する命令です。

■表示、プリント機能
▶PMODE：この命令により、MZ-80K/C/1200用のプリンタの他、MZ-80B/2000用、セントロニクス形式のプリンタが使用できます。(キャラクタ・セットの違いにより、MZ-80K/C/1200用プリンタ以外は、特殊な文字はプリントできないことがあります。)

■その他
▶キーボードの先行入力、オートリピート、2キーロールオーバーが可能になりました。▶16進数がどこにでも書けるようになりました。▶ストリングの大小比較が可能です。このため、文字やアルファベットの順番の並べ換えが簡単に行なえるようになりました。▶画面表示がスピーディーになりました。▶ファイル処理が高速化されました。特に、BRDファイルでは、平均4～5倍の高速処理を実現します。▶その他、IF～THEN-ELSE,AND、OR、NOT、PRINT USING、AUTO、RENUM、INSTR,WAIT、BEEP、HCOPY、などが追加され、DIM、CHR$、POKE、USRなどの命令が機能アップしました。

●10進10桁演算でビジネス用にも安心してご利用できます。※表示桁数は、変更できます。●カセットテープや単密度(SP-6010、SP-6110、SP-7010)のファイルは、内蔵のコンバート・ユーティリティによって、倍密度ディスクに転送できます。この結果、これまでのプログラムやデータをムダにすることがありません。

待望の新発売！

MZ-1200/K/C用
Telly-BASIC(テープ版、10進10桁演算)　BAS-10T
標準価格　3,800円

●シャープSP-5030とテキスト・コンパチブルです。BAS-10Dのテープ版です。※SP-5030に機械語を組合わせたものには、働かないことがあります。●多彩な拡張能力(BAS-10Dのうちディスク処理コマンドを除いたすべての能力)が、魅力です。●わかりやすいマニュアルつきで、何と3,800円です。

発売まであとわずか！

MZ-1200/K/C用
Telly-BASIC(テープ版、高速整数演算)　BAS-11T
標準価格　3,800円

●BAS-10Tを整数形にしたものです。だから高速です。※整数範囲は、−32768～32767です。●ふんだんに盛込まれた拡張機能がバツグンです。●ハイスピードゲームに最適！※㈱ロータスのゲームボードをサポートしていますから、すごいゲーム感覚が体験できます。

㈱ロータスと技術提携

●MZのグラフィックを強化したゲームボードを開発・販売しているロータスと技術提携しました。

《マニュアルだけの販売も行なっております。》

●FDC-10H、その他Telly-BASICシリーズのうち、ご希望の品名を明記し、"マニュアルのみ"とご注文ください。
標準価格　各1,500円

只今開発中！

●テープBASICコンパイラ(MZシリーズ)●ディスクBASICコンパイラ(MZシリーズ)●CP/M-FDOS交換プログラム(MZ-80B/2000FDOS用)●その他、各種MZサポートプログラムを開発予定です。

通信販売のお知らせ(送料無料)

- ●ご希望の商品価格の合計金額を、「現金書留」「郵便振替」、または「銀行振込」にてお申込みください。
- ●お名前、住所、電話番号、商品名は、ハッキリご連絡ください。
- ▶郵便振替口座：大阪9-310106
- ▶銀行振込口座：住友銀行日本一支店・普通：No.709565

〒556 大阪市浪速区下寺町2-2-16
☎06—631—0925

代理店募集中
●お気軽にお問い合わせください。

Hudson soft

HuBASIC™

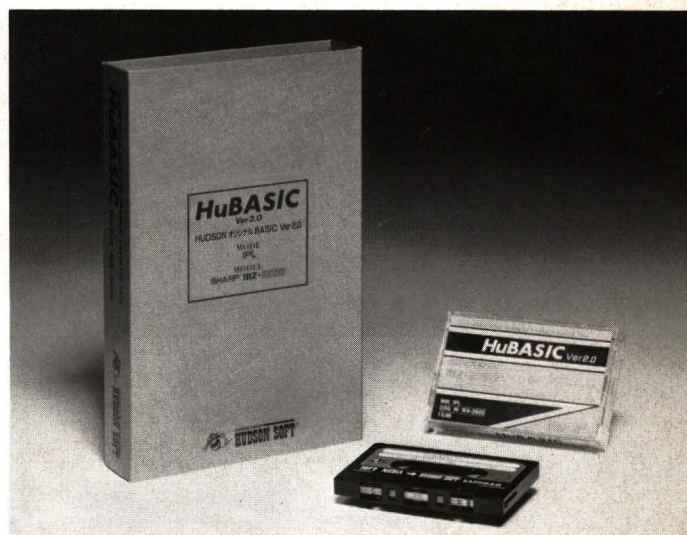
堂々完成！

HuBASIC Version 2.0

HUDSONオリジナルBASIC Ver 2.0

MODEL:SHARP MZ-2000　MODE:IPL

Price:¥10,000　Serial WA-3600

カセット・マニュアル付

MZ-2000 HuBASIC Ver2.0の豊富な機能

1）200種を越える豊富な命令群

2）MZ-2000のカラー機能をフルサポート
タイリング・ペイント，点線，破線，WINDOW命令による強力なカラーグラフィック。

3）命令語の省略が可能(例 RUN→R., LIST→L.)

4）ファイル・ディスクリプタによる自在なファイル管理が可能
4Mバイトまでの範囲内で外部メモリーの設定が可能
㈱I・Oデータ機器発売中の大容量RAMボード（PIO-3034(320Kバイト))★を組み合わせることによって大量のデータ処理，プログラム処理が可能となります。
なお，グラフィックボード(MZ-1R01)を使用してのファイル，データ処理も可能です。

5）漢字出力機能
同じく㈱I・Oデータ機器発売中の漢字ROMボード★★を組み合わせることによってMZ-2000が簡単に漢字を出力することが出来ます。
カナ⇨漢字コード変換機能もサポートしています。

6）演算精度は整数，単精度８桁，倍精度16桁を持ち，かつ倍精度演算による関数計算もできます。

7）グラフィック・ウィンドウの座標系の拡張
±38乗以内の範囲で設定できます。

8）演算スピードをさらに高速化
演算スピードはVersion1.シリーズより高速化を計り，さらに各ステートメント自体の実行スピードもアップさせています。

9）ロングバリアブルネーム(Long Variable Name)の採用 240文字までの変数名が使え，かつアップバーや英小文字も変数に使えますので,REPEAT-UNTIL,WHILE～WENDなどの構造化プログラミングに適した命令や字下げを有効に使うことによってプログラムの保守性を向上させることができます。

10）強力なスクリーンエディタによるプログラム編集効率の向上、変数を保護しながら編集可能，さらにプログラムをチェーンしても変数は保護されます。

11）構造化プログラミング命令の採用
FOR～NEXT,REPEAT～UNTIL,WHILE～WEND,LABELの各命令を持っています。

12）パソコンTV X1，MZ-700，MZ-80B(HuBASIC ver2.0管理下)の各機種のソフトウェアを読むことができます。
(ただし各機種特有の機能はサポートされません)。

13）キーファンクションの大幅な採用

14）デバッグ用トレース機能の装備

15）PRINT USING命令の装備

16）近日発売予定のディスクベーシックとは完全なコンパチビリティを保っています。

17）MZ-2000，HuBASIC ver2.0を使用して外部メモリーに倍精度数値データを格納した場合。
MEM:グラフィックメモリ（MZ-1R01にメモリーをフル実装48Kの場合）
5119ヶのデータを格納。
EMM:320KバイトRAMボード(PIO-3034)の場合
39935ヶのデータを格納できます。

```
10 SCREEN0,0:OPTIONSCREEN2:INIT"MEM:"
20 CLS:INPUT"VDIM NAME=",F$:INPUT"VDIM ｻｲﾀﾞｲ ﾖｳｿ ｽｳ=",MX:M=INT(MX/32)
30 OPEN"R",1,"MEM:"+F$:IFLOF(1)=0THENFORI=0TOM:PUT#1,I:NEXT:GOTO50
40 IFLOF(1)<M THEN CONSOLE:CLOSE:ERROR10
50 GET#1,0:X=-1:CONSOLE3,22
60 DIM BF$(31):FORI=0TO31:FIELD#1,I*8ASAA$,8ASBF$(I):NEXT
70 X=X+1:INPUT"ﾋｷｽｳ ﾉ ｱﾀｲ(-1 ﾃﾞ END)=",X:IFX<0GOTO110 ELSEGOSUB120:'D#=VDIM(X
80 PRINT"VDIM(";X;") ﾉ ｱﾀｲ=";D#:N#=D#
90 INPUT"ｶｷｶｴﾙ ﾃﾞｰﾀｰ",N#:IFD#<>N#THEN D#=N#:GOSUB130:'VDIM(X)=D#
100 PRINT:GOTO70
110 PUT#1,LOC(1):CLOSE:CONSOLE:END
120 GOSUB140:D#=CVD(BF$(G2)):RETURN
130 GOSUB140:BF$(G2)=MKD$(D#):RETURN
140 IFMX<X THEN CONSOLE:CLOSE:ERROR9
150 G1=INT(X/32):G2=X-G1*32:IF G1=LOC(1) THEN RETURN
160 PUT#1,LOC(1):GET#1,G1:RETURN
```

注１ ★ PIO-3034，MZ-2000/80B用，320キロバイトRAMボード￥98,000
MZ-2000/80Bの拡張I/Oユニットに納まる，I/Oポート接続の320KB RAMボードです。Z-80AのI/O命令にて，アクセスできフロッピー，フロッピーの増設代替としても使用できます。
用途：画像データのバッファ／計測データのバッファ／FDD代替。

注２★★PIO-3055，MZ-2000/80B用，漢字ROMボード ￥35,000
MZ-2000/80Bの拡張I/Oユニットに納まる，漢字パターン書き込み済ボードJIS第一水準＋αの3700文字のほか，ユーザー作成の漢字，マーク等 500字を追加できる特徴を持ちます。
(16×16ドット漢字3700字＋2764用EPROMソケット付)

注1，注2開発・販売元 ㈱I・Oデータ機器
〒920 石川県金沢市高岡町7-22 TEL.0762(21)4812

お問い合せは、最寄りのハドソン各営業所へ

堂々完成
ラインナップも豊富に

最高級ビジネス用
簡易言語の決定版

カセット

MODEL	SERIAL	MODE	PRICE
MZ-80K/C. K2. K2E. MZ-1200.	Z-3800	MONITOR	¥8,800
MZ-80B. B2	W-3800	IPL	¥9,800
MZ-2000	WA-3800	IPL	¥9,800
MZ-700	QA-3800	MONITOR	¥8,800

ディスク

MODEL	SERIAL	MODE	PRICE
MZ-80B	W-3850-D	IPL	¥19,800
MZ-K C. K2. K2E. 1200	Z-3850-D	MONITOR	¥19,800
MZ-2000	WA-3850-D	IPL	¥19,800

概要紹介

HuCALは表形式のデータベースシステムです。表の横方向を列、縦方向を行といいます。表は横方向にA、B、C……、Y、Z、AA、AB、……、IT、IU、の255列、縦方向に0から10000の10001行あります。実際には、記憶容量に制限されるので(約40KB)、HuCALを起動した時点で、最大8000行がデータ入力の限度となります。

表はフィールドと呼ばれる単位データによって構成されます。フィールドの指定はAの100（A列目の100行目）というようにされますが、式やマクロ命令のなかでの指定には！（A，100）という記号が用いられます。またAからIUは0から254までの数値と同等に扱えます。従って、！（A，100）と！（0，100）は同じフィールドを指定します。各フィールドには数値、文字、式、マクロ命令のいずれでも入力することができます。各フィールドの長さは3ケタから74ケタまで自由に変えることができます。長さの変更は、フィールドにすでにデータが登録されているときでも行えますが、数値のケタ数が、フィールドの長さより大きくなると、フィールド全体に%記号が表示されます。また1つのフィールドに入力できる文字数は最大74文字です。マクロ命令は、0列0行から20列20行までの合計441個のフィールドの範囲で、239文字まで入力可能です。各フィールドにはデータ保護のためにプロテクトをかけることができます。

HuCALの式やマクロ命令には15種類の関数を使用することができます。関数にはSUM（合計計算）やMAX(最大値）などが用意されており、従来のBASICではループ計算で行っていた計算が簡単に命令できます。また、マクロ命令には文字列の比較機能が用意されていますから、顧客管理のようなデータ検索も簡単に行えます。マクロ命令のコマンドは9種類だけですので、すぐに覚えて使いこなせるでしょう。マクロ命令の内部にはラベルをつけることができますので、BASICのFOR～NEXT文に相当するループ文を書くこともできます。さらに、あるマクロ命令の実行が終った時点で他のマクロ命令に実行を移すことも可能です。

主な特徴

◎16桁の高精度計算
◎倍精度関数を装備
◎フルスクリーンエディターによる編集機能付
◎フォーマット指定が容易
◎フィールド調整はカーソルキーでOK

ハドソン札幌/〒062 札幌市豊平区平岸3条5丁目4番17号 コロナード平岸II201 PHONE:011-821-1538
ハドソン大阪/〒542 大阪市南区南船場4丁目2番18号 佐野屋ビル5F PHONE:06-251-1945
ハドソン東京/〒102 東京都千代田区麹町4丁目7番5号 麹町ロイヤルビル2F PHONE:03-234-4996
ハドソンUSA/2063 CENTER STREET BERKELEY CA 94704 TELEPHONE 415-845-1416

新作　　　　ゲーム

Hudson soft

おもしろゲームソフト

●パソコンテレビX1

ソフト名	SERIAL		言語	PRICE(¥)
スペース・ハンバーガー	X-1001-G	アッ！ハンバーガーが飛んできた！	M	3,800
ベジタブル・クラッシュ	X-1002-G	にんじん君、なす君、りんご君が登場！	M	3,800
モンキー・アップ	X-1004-G	おサルの投げるおにぎりをキャッチ、ウメボシ爆弾でやっつけろ。	Hu	3,200
スーパー・ゴルフ	X-1006-G	コースを回る休日のゴルフは楽しいものです	Hu	3,800
F1レーサー	X-1008-G	さあ、君は今からプロのドライバー	Hu	3,800
利息のガイド	X-1009-G	いま、あなたは銀行にいくら預金していますか	Hu	3,200
立体バレーボール	X-1010-G	本格的なバレーボールゲームの登場です	Hu	3,800
ホースレース	X-1011-G	君は何レースにかけるかな？	Hu	3,800
さすらいの忍者	X-1012-G	敵の忍者にみつかる前にたくさんの巻物をみつけよう	Hu	3,800
かえるがかえる	X-1013-G	迷子のかえるをおうちに返して下さい	Hu	3,800
初級英単語レッスン	X-1014-G	マイコンで英単語を記憶しよう	Hu	3,200
成績処理	X-1015-G	学校で試験の成績を計算するソフト	Hu	3,200
占星術	X-1016-G	すばらしい星占いの世界へ御招待します	Hu	3,800
姓名判断	X-1017-G	字数、字画数で姓名判断します	Hu	3,800
君はピアニスト	X-1018-G	X1で優雅なメロディを奏でましょう	Hu	3,200
俺はバイキング	X-1019-G	暴れるバイキングを大砲でやっつけよう	Hu	3,200
エスケープ大作戦	X-1020-G	先生の目をぬすんでエスケープ	Hu	3,800
スーパーバリケード	X-1021-G	バリケードの強化版	Hu	3,200
スナイパー	X-1022-G	目標を一撃のもとに倒すのがスナイパーの使命なのだ。	Hu	3,200
頭の体操	X-1023-G	あなたはどの程度の頭脳を持っているか？	Hu	3,200
ピンボール	X-1025-G	アメリカ版のパチンコです	Hu	3,800
ローディボール	X-1026-G	移動寿司＋ブロックククズシの超ウルトラゲーム	Hu	3,200
タイガー・コンバット	X-1028-G	宇宙戦車のパイロットは君だ！	Hu	3,200
ザ・スパイ	X-1029-G	あなたは弾より速い、スーパースパイ	Hu	3,800
来なさい！	X-1030-G	エイリアンが家の中に大挙して攻めてきました	M	3,800
爆弾男	X-1031-G	風船オバケをやっつけろ	M	3,800
クレイジースパイ	X-1032-G	猛烈な敵宇宙船の攻撃を突破できるか！	M	3,200
キャノンボール	X-1033-G	さあ、大変！タマの洪水だ	M	3,800
夢のプロ野球	X-1034-G	巨人VS西武、夢のカードが実現	B	3,200
あなたは名カメラマン	X-1035-G	森の動物写真を撮りましょう	M	3,200
カエルシューター	X-1036-G	怪物カエルが襲ってくるぞ	M	3,800
ガンマン	X-1037-G	君は西部のプロガンマン	M	3,800
サブマリンパニック	X-1038-G	敵潜水艦は近いぞ！	M	3,200
グルグル	X-1039-G	目鼻、口のオバケがグルグル？	M	3,200
ねずみとり	X-1040-G	君は何匹、ねずみを退治できるか	M	3,200
スプリング・バグ	X-1041-G	神殿にいる伝説の娘に出会えるかな？	M	3,800
ぐしゃ	X-1042-G	好物のチーズをうまくとって逃げるのだ！	M	3,200
ミスターバタフライ	X-1043-G	あっ！蝶々が襲ってくる	M	3,800
ヒヨコ・ファイター	X-1044-G	ヒヨコになってタマゴを食べよう！	M	3,200
スクウェアーガーデン	X-1045-G	魔の部屋は熾烈な闘技場だ	M	3,800
ブーメラン・ハンティング	X-1046-G	必殺ブーメランで獲物を採りましょう	M	3,800
4人マージャン	X-1047-G	グラフィックによる本格的4人麻雀	M	4,800
野球拳	X-1048-G	ちょっと色っぽいおなじみよよいのよい	M	3,200
詰将棋	X-1049-G	これで君も将棋のプロになれるぞ	M	4,800
マトリックス会計	X-3001-G	マトリックス・チャートにより財務情報を出力します	Hu(O)	3,200
給与計算	X-3002-G	給与明細から金種計算までOK	Hu(O)	3,200
Zチャート 季節変動指数	X-3003-G	過去3年間の売上データで、今後の販売予測をたてます	Hu(O)	3,200
家計簿	X-3004-G	家庭の主婦のお手伝い	Hu(O)	3,200
DATA BASE	X-3051	すべての情報ファイルはこれでOK！	M(O)	3,800

※HuはHuBASIC　●SはSharp Basic　●Mはマシン語。

●MZ-700

ソフト名	SERIAL		言語	PRICE(¥)
ダイヤモンドコブラ	QA-1002	迷路の中のダイヤをさがせ	M	3,200
キャノンボール	QA-1021	さあ、大変！タマの洪水だ	M	3,200
スーパーゴルフ	QA-1003	コースを回る休日のゴルフは楽しいものです	Hu	3,800
ムーンサルト	QA-1004	うまく月面に軟着陸できるか？キミの逆噴射次第	S	2,800
カンニング大作戦	QA-1005	期末試験でカンニングしている生徒に注意！	S	2,800
緊急発進	QA-1006	UFOの攻撃を防げるか！1秒を争う時間との勝負	S(O)	2,800
ホースレース	QA-1007	君は何レースに賭けるかな？	Hu	3,200
エアーバトル	QA-1008	飛来するUFOをミサイルで撃滅せよ	M	2,800
バレーボール	QA-1009	立体バレーボール、1人でも3人でもできます	Hu	3,200
スペースシューティング	QA-1010	照準を適確に定めてミサイルON！	Hu	2,800
シミュレーション＃1 病院経営ゲーム	QA-1011	あなたは病院を経営できるか？	Hu	2,800
シミュレーション＃2 あなたは殿様？	QA-1012	エゾの国の経営はあなたの釆配しだいです	S	2,800
頭脳ゲーム	QA-1013	皆でやってみよう頭の体操	Hu	2,800
リバース	QA-1014	マイコンを打ち負かすことができるか	S	3,200
戦国軍団	QA-1015	おかしなおかしな戦争ゲーム	M	3,200
HuPACK＃1 マージャン・雀球	QA-2001	振り込みに注意しながら、半チャンをやりましょう あなたのコインの数は？腕前が試されますよ	Hu	3,200
HuPACK＃2 ローディボール・野球拳	QA-2002	ボールをうまくブロックに当てて、高得点を狙ってください ちょっと色ぽい、おなじみよよいのよいゲーム	Hu	3,200
HuPACK＃3 アリゲータ・パチンコ	QA-2003	うまくジャンプして、ヤシの実を取ってください さて、あなたは台を定量にできるかな？	Hu	3,200
HuPACK＃4 スタートレック・レイホープ研究所	QA-2004	マイコンゲームの古典的名作 研究員の救出隔離をしなければなりません	Hu	3,200
HuPACK＃5 ノルマンディー・ラスベガススロット	QA-2005	史上最大の作戦ノルマンディー上陸作戦をくい止めよう おなじみのスロットマシンゲーム	Hu	3,200
HuPACK＃6 ボウリング・ベースボール	QA-2006	マイコンでボウリングをしよう マイコンベースボールの興奮を！	Hu	3,200
HuPACK＃7 ブラックジャック・ポーカー	QA-2007	カードの合計を21に近づけるゲーム 1人で楽しむポーカー	Hu	3,200
HuPACK＃8 バリケード・陣取りゲーム	QA-2008	画面上の星をいくつ確保できるか 相手に陣地を与えないよう迅速・敏感に判断せよ	Hu	3,200
S-PACK＃1 さるも木から落ちる・ハングマン	QA-2100	4本の木の間をさるが飛びかいます 死刑囚を救うのはあなたの頭しだい	S	3,200
爆弾男	QA-1016	風船オバケをやっつけろ	M	3,200
ガンマン	QA-1017	君は西部のプロガンマン	M	3,200
夢のプロ野球	QA-1018	巨人VS西武夢のカードが実現！	Hu	3,200
ぐしゃ	QA-1019	好物のチーズをうまくとって逃げるのだ！	M	3,200
サブマリン・シューター	QA-1020	アトランテス大陸の謎へチャレンジ！	M	3,200
HuCAL	QA-3800	最高級ビジネス用簡易言語	M(O)	8,800
顧客管理	QA-3001	何にでも使える顧客管理、商店用に最適	M(O)	3,000
在庫管理	QA-3002	毎日の入出庫管理	S(O)	3,000
ビジネスパック＃1 クレジット計算・金種計算	QA-3003	銀行金利やローン返済の計算 給料日の経理課のお手伝い	S	2,800
ABC分析	QA-3004	入力データの構成比の高い順から色分けで出力	Hu(O)	2,800
Zチャート 季節変動指数	QA-3005	過去3年間の売上データで今後の販売予測をたてます	Hu(O)	2,800
DATA BASE	QA-3100	すべての情報ファイルはこれでOK！	M(O)	3,200

速報！

●MZ-2000用最新作

本格的シュミレーションゲーム登場

BATTLE OF BRITAIN

WA-1058-G　言語B　¥4,800

速報！

※最新ソフトカタログをご希望の方はX1, MZ-700, FM-7, MZ-2000, PC-8001mkIIの機種を明記の上ハドソン東京まで〒60円を添えて、お申し込み下さい。

※通信販売はソフト名、SERIAL No.を明記の上お近くのハドソンZ-7係までお送り下さい。

※NO表示のGはグラフィックRAMが必要です。※言語欄に(O)印のあるものはマニュアル付き

あらゆるメディアに人間性を

HUDSON GROUP HUDSON SOFT

ハドソン札幌/〒062 札幌市豊平区平岸3条5丁目4番17号 コロナード平岸II 201 PHONE:011-821-1538
ハドソン東京/〒102 東京都千代田区麹町4丁目7番5号 麹町ロイヤルビル2F PHONE:03-234-4996
ハドソン大阪/〒542 大阪市南区南船場4丁目2番18号 佐野屋橋ビル5F PHONE:06-251-1945
ハドソンUSA/2063 CENTER STREET BERKELEY CA 94704 TELEPHONE 415-845-1416

SYSTEMコーナー
SOFTコーナー
PLAYコーナー

メディア札幌

■所在地／札幌市中央区南１条西２丁目
■TEL／(011)281-1151(内線2294)
■営業時間／AM.10：00～PM.6：00
（土、日曜日はPM.6：30まで）
■定休日／毎週水曜日

ハドソンソフト直営店

道内最大規模北海道パソコンセンター内

シャープパソコンランド札幌　　富士通マイコンスカイラブ札幌
東芝パソコンサロン札幌

ボーナスでビッグチャンス

差をつけるのは今だ！

ボーナス兼用で支払楽々

MZ-731セット
〈ビジネスマンコース〉

品目	価格
MZ-731	¥128,000
カラーディスプレイ14M-111C	¥ 67,800
ケーブル	¥ 1,800
ソフト10本	¥ 32,000
SP-5030	¥ 3,000
ロールペーパー	¥ 490
ボールペン黒又４色	¥ 850
本（パソコンサンデー）	¥ 980
（アタッシュケース） 7,000円プラス	
合　計	¥231,920
販売価格	**¥198,000**

ローン 頭金	1回目	2回目以降	ボーナス
20回 0円	8,840円	7,100円	20,000円×4回

MZ-721セット
〈入門者コース〉

品目	価格
MZ-721	¥ 89,800
カラーディスプレイ14M-111C	¥ 67,800
ケーブル	¥ 1,800
ソフト10本	¥ 32,000
本(パソコンサンデー)	¥ 980
SP-5030	¥ 3,000
合　計	¥195,380
販売価格	**¥158,000**

ローン 頭金	1回目	2回目以降	ボーナス
20回 0円	6,440円	5,900円	15,000円×4回

※今、MZ-700シリーズ御買上のお客様にロータスゲームボードを２割引で販売します!!

パソコンテレビX1セット

〈入門セット〉

品目	型番	価格
本　体	CZ-800C	¥155,000
ディスプレイ	CZ-800D	¥118,000
G-RAM	CZ-8GR	¥ 32,000
ハドソンゲームソフト10本		¥ 32,000
JOYスティック		¥ 3,900
私の勉強ノート（ラジオ技術社刊）		¥ 1,500
合　計		¥337,400
販売価格		**¥300,000**

ローン 頭金	1回目	2回目以降	ボーナス
36回 0円	9,200円	8,600円	10,000円×6回

〈漢字ワープロセット〉

品目	型番	価格
入門セット		¥337,400
拡張I/Oポート	CZ-8EP	¥ 11,800
漢字ROM	CZ-8KR	¥ 38,000
ワープロソフト(HuWP)		¥ 24,800
プリンター	CZ -80P	¥142,800
合計		¥554,800
販売価格		**¥492,600**

ローン 頭金	1回目	2回目以降	ボーナス
36回 0円	10,368円	8,500円	50,000円×6回

富士通

FM-7セット

〈入門者コース〉

品目	価格
FM-7	¥126,000
カラーCRT	¥ 99,800
ケーブル	¥ 1,800
ソフト10本	¥ 32,000
データーレコーダー	¥ 12,800
合　計	¥272,400
販売価格	**¥238,000**

ローン 頭金	1回目	2回目以降	ボーナス
36回 0円	9,692円	6,400円	10,000円×6回

〈上級者コース〉

品目	価格
FM-7	¥126,000
カラーCRT MB27303	¥ 99,800
ケーブル	¥ 1,800
ソフト10本	¥ 32,000
プリンター MB27405 プリンター用紙サービス	¥ 89,000
ケーブル	¥ 4,500
ミニフロッピー	¥ 98,000
〃 インターフェース	¥ 4,800
フロピー10枚	¥ 17,000
システムフロッピー	¥ 7,000
合　計	¥489,900
販売価格	**¥427,600**

ローン 頭金	1回目	2回目以降	ボーナス
36回 0円	15,036円	11,700円	30,000円×6回

東芝

パソピア7セット

品目	型番	価格
パソピア7	PA7007	¥119,800
ファインカラーディスプレイ	PA7165	¥ 98,000
ハドソン製ゲームソフト10本		¥ 19,000
データレコーダー	当社推奨品	¥ 12,800
合　計		¥249,600
販売価格		**¥230,600**

ローン	1回目	2回目以降	ボーナス
36回	8,060円	7,900円	なし

あらゆるメディアに人間性を
HUDSON GROUP
HUDSON SOFT

ハドソン札幌／〒062 札幌市豊平区平岸3条5丁目4番17号 コロナード平岸Ⅱ201 PHONE:011-821-1538
ハドソン東京／〒102 東京都千代田区麹町4丁目7番5号 麹町ロイヤルビル2F PHONE:03-234-4996
ハドソン大阪／〒542 大阪市南区南船場4丁目2番18号 佐野屋橋ビル5F PHONE:06-251-1945
ハドソンUSA/2063 CENTER STREET BERKELEY CA 94704 TELEPHONE 415-845-1416

誰にでも手軽に使える

高級ソフトウェアの開発・販売

安くなったコンピュータを専門知識がなくても手軽に実務に役立てるようにしたい。これが当社の願いです。

MZ-80B
2000用

●汎用実務ソフトウェアの優劣は
①各種、各規模の会社にどれだけの適用性があるか……機能、
②操作ミスに対してどれだけの防御をプログラムに組み込まれているか……品質、によって決まります。特に②の要素はコンピュータを有効利用する上で極めて重要です。

●コンピュータを熟知した会計士数名のノウハウを結集したシステムです。

※(D)はディスケット、(T)はテープ

※高級財務会計システム(D)　ABC-16 ¥150,000

(専用帳表あり)
データ量／2800仕訳／月　科目数／244種
入力伝票／入金、出金、振替、複合振替
科目入力／ワンタッチ・キー入力またはコード3桁入力
帳　表／日記帳、合計残高試算表、総勘定元帳、決算報告書、仕訳リスト、仕訳訂正・取消リスト、マスター登録リスト
画面表示／マスター登録データ、残高、仕訳データ、預金口座別入出金合計、残高試算表(B/s、P/L)
訂正・削除／入力時仕訳データ、FD内仕訳データ、マスターデータ、残高
その他／締切年月日は自由に設定可。月末・期末自動更新、仮締プリント、元帳科目指定プリント可

※簡易財務会計システム(D)　ABC-10 ¥50,000 応用紙タイプ　ABC-11 ¥60,000 専用紙タイプ

ABC－16の簡易形システムで、月次の処理ができます。
入力は振替伝票形式で科目コードを用いて入力。
帳　表／日記帳、合計残高試算表、総勘定元帳、仕訳リスト、仕訳訂正・取消リスト、マスター登録データリスト

※高級財務会計システム(D)　ABC-12 ¥120,000 応用紙タイプ　ABC-13 ¥130,000 専用紙タイプ

ABC－16と簡易システムの中間グレード、決算報告書の作成まで。

■MY-REPO-180　(T) ¥19,800　(D) ¥29,800

横項目の四則演算、縦の大、中、小の中間計が自由に取れます。
20項目の複合検索ができます。
計算結果の数値データの小数点の位置指定、数値データはカンマの有無指定もできます。宛名プリント、棒グラフ、作表等の多目的簡易言語

■顧客管理システム(D)　ABC-62 ¥30,000

販売戦略に有効活用する。1,000名(IFD)の顧客名簿の管理。顧客マスターに〒NO、人名等を登録し各種の条件により宛名ラベル、該当者リストをプリント。検索方法は、地区、業種、8桁のステータスコードを組合せ自在の条件にて検索。

■高級比較棒グラフ(D)　T-11 ¥30,000

500項目の月別計画または、前年度実績等をマスター登録し、実績データの入力により各種の比較棒グラフをプリンターに出力。12ヵ月分のデータは計画と実績、いずれも保存されますので任意のデータ処理が可能。
作　表／項目(4桁)別該当月対比グラフ、合計グラフ、総合計(実数)、累計グラフ、月別推移グラフ、グループ(コード上2桁)別の同上グラフ　**高級比較棒グラフ(T) ¥15,000 (120項目)**

●来社テスト自由
●カタログ請求券と〒200円分を同封の上当社までご請求下さい。
●ソフトのお求めは、お近くの有名百貨店、ショップ店等の業務用ソフト取扱い店かまたは直接当社宛に現金書留か銀行振込みでお願い致します。

※売掛金管理システム(D)　ABC-30 ¥70,000

データ量／2,000件／月　得意先／160社
入力区分／売上、入金、値引戻り、相殺勘定
取引先の入力／ワンタッチ・キー＋3桁コード入力
締切区分／8種
帳表類／売上入金日報、得意先元帳、売掛残高一覧表、請求残高一覧表、請求明細表、マスター登録データリスト
画面表示／マスター登録データ、残高、入力データ一覧
訂正・削除／マスター登録データ、残高、入力データ
更新処理／月末、期末自動更新

※買掛金管理システム(D)　ABC-40 ¥50,000

売掛金管理ABC－30と同機能あり。

※給与計算システム(D)　ABC-50 ¥50,000

(専用帳表使用)
対象人員／200名(IFD)　給与区分／月給、日給、時間給
支給項目／14種 (内7項目は自由に設定)
控除項目／13種 (内8項目は自由に設定)
税区分／甲種、乙種
その他／部門別(10部門)集計、有給管理、特別休暇の処理
帳表類／賃金台帳、給与明細書(以上専用紙)、支給一覧表(部門別、合計)、控除一覧表(部門別、合計)、地方税納付先別一覧表、金種表

※賞与計算システム(D)　ABC-51 ¥40,000

(専用帳表使用)
ABC－50給与計算連動式
賞与支給額の算定用試案表ができます。
帳表類／賃金台帳、賞与支給明細書(以上専用紙)、支給一覧表、控除一覧表、賞与試算表、金種表

※年末調整システム(D)　ABC-52 ¥30,000

帳　表／個人別源泉徴収簿、合計表、一覧表、総括表

■定期刊行誌管理(D)　ABC-60 ¥130,000

契約購読料の入金、請求、契約更新、刊行誌の発送等合理化。

■計算集計(T)　T-01 ¥10,000

<R.　P.　G> 横7項の四則演算及び60項目の合計。
<6ヶ月サマリー> 横6項目縦225項目の各平均、合計。

■グラフ作成(T)　T-10 ¥10,000

比較棒グラフ、単純棒グラフ、折線グラフ、X-Yグラフをプリンターに。

■グラフ作成(T)　T-15 ¥6,000

18項目の数値を棒グラフ、点グラフにして画面、プリンターに。
※マークはK/C1200用も用意してありますが機能面の差が多少あります。

ABC SYSTEM ピーシーエー株式会社

ピーシーエー株式会社

東京本社　〒150 東京都渋谷区渋谷3-15-5　グリームビル5F
TEL 03-406-3290㈹　振込口座 富士銀行渋谷支店普通No.1412047
大阪本社　〒530 大阪市北区中崎西2-2-1 東梅田八千代ビル3F
TEL 06-315-8637㈹ 振込口座 富士銀行大阪駅前支店普通No.796954
名古屋本社　〒451 名古屋市西区菊井2丁目6-7　TEL 052-565-1527

マイコン

有名メーカー

店頭発表価格奉仕中！

チャンス！

ソフト2000本

クレジットOK

7月末
マイコンフェア
開催予定

マイコン・ショップ1周年セール中！

NEC

PC-8001mkII ……………… ¥~~128,000~~

ベストセラー機PC−8001の使い易さをそのまま活かし、従来の豊富なソフトウェアをそっくり使えて、ビジネスからホビーまで活躍します。

PC−9801	¥~~298,000~~
PC−8801	¥~~228,000~~
PC−8201	¥~~138,000~~
PC−6001	¥~~89,800~~
PC−2001	¥~~59,800~~

SHARP

パソコンテレビX1

パーソナルコンピュータ＋キーボード
CZ-800C ¥~~155,000~~
14型カラーディスプレイテレビ………
CZ-800D ¥~~113,000~~
パソコンテレビ……………………………
CZ-800C/D ¥~~268,000~~

●コンピュータ・グラフィックス●ビデオ編集●コンピュータ学習●シンセサイザーミュージック●ワードプロセッシング●ホームコントロール●テレビゲーム●テレビコントロール●ビジネス計算

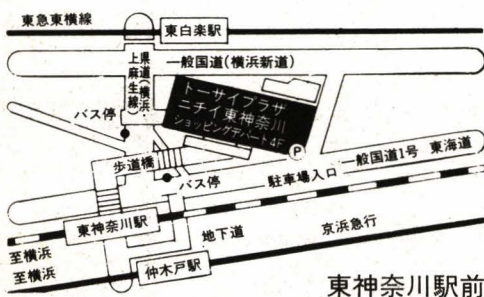
MZ−2000	¥~~218,000~~	PC−1251	¥~~29,800~~
MZ− 700	¥~~79,800~~	PC−1250	¥~~22,800~~
PC−1500	¥~~59,800~~	PC−1245	¥~~17,800~~

富士通

FM-7（簡易言語ソフト付）
¥~~126,000~~

タモリでお馴みFM−7、簡易言語付ですから、購入したその日から、幅広い用途に活用できます。

FM−11EX(簡易言語付)……	¥~~398,000~~
FM−11AD(簡易言語付)……	¥~~338,000~~
FM−11ST…………………	¥~~268,000~~

カシオ	エプソン
FP シリーズ 3割引	フロッピーディスク プリンタ各種
	その他多数

■お支払いは便利なクレジットで
ボーナス一括払い実施中！

●お支払い例（¥215,000お買上げの場合）
国内信販でご利用の場合
月々わずか ¥23,650×10回
月々わずか ¥12,680×20回

●住友カード●ダイヤモンド●ユニオン●日本信販カード●ミリオンカード●日本ダイナーズ●ＪＣＢ……………取扱店

ペロトーカイプラザ ニチイと70の専門店・4F

横浜市神奈川区富家町１番地（東神奈川駅前）TEL.045(421)3005

時代が変わる。渋谷が変わる。

4F

パソコン教室

■パソコン教室
■パソコンプレイルーム

パソコン教室では、BASICの初級から上級までの各コースのほかに、マシン語活用コース、ワープロコース、簡易言語活用コース等、盛りたくさんのコースを開講しあなたのコンピュータライフをサポートします。

パソコン教室 6月・7月のスケジュール

コース名		開講日	時間	受講料
BASIC基礎講座	初級コース	7月3日㊐・10日㊐・17日㊐	AM11:00～PM6:00（6時間）	5.000円
	中級コース	7月2日㊏・16日㊏		7.000円
	上級コース	7月9日㊏		8.000円
マシン語講座	Z80初級コース	6月25日㊏	AM11:00～PM6:00（6時間）	5.000円
活用講座	簡易言語活用コース	7月1日㊎・15日㊎	AM11:00～PM5:00（5時間）	5.000円
	ワープロ活用コース	6月24日㊎・7月8日㊎・22日㊎		
	CP／M入門コース	7月7日㊍・21日㊍		
プレイルーム		毎週 月・火曜日	AM11:00～PM7:00	300円(1時間)

3F

OA機器・専門書籍

■ビジネスパソコン
■ワードプロセッサ
■ビジネスソフト
■専門書籍
■ハンドヘルドコンピュータ

各メーカーのビジネスパソコン、ワードプロセッサ等のOA機器の専門フロアーです。

OA機器

OA機器コーナーではワードプロセッサ、ファクシミリなどOA化には欠かせない機器を取揃えています。

ビジネスソフト

J&P自慢のソフトコーナーは豊富なアプリケーションがギッシリ！

2F

ビジネスのパソコン

■パソコン
■ディスプレイ
■プリンタ
■フロッピィディスク
■パソコンアクセサリー

各メーカーの話題のパソコン、周辺機器を大量に取揃えたワイドなフロアーです。

世界のパソコン

国内・海外の人気機種がズラリ勢揃い。目的に合った機種がお選びいただけます。

周辺機器

パソコンの機能を拡げるディスプレイ、プリンタなどを集中展示しております。

1F

ホビーのパソコン

■ホビーパソコン
■テレビゲーム
■ゲームソフト
■学習ソフト

ホビー、ゲームのためのパソコン、初心者のためのパソコンを各種取揃えています。どなたにでも、パソコンをエンジョイしていただける楽しいフロアーです。

ゲームソフト

全国から集めた優秀なソフトが3,000種類。お求めやすく機種別の展示が好評です。

ホビーのパソコン

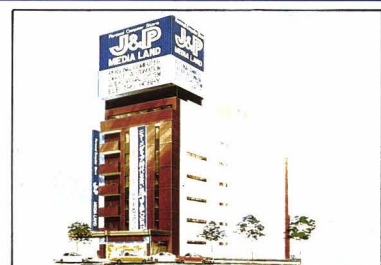
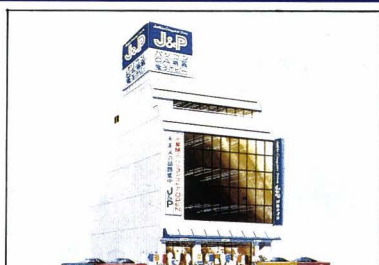
ホビィストには見逃せない楽しさいっぱいのコーナーです。

J&Pアドバイザー 小松左京

J&P渋谷店

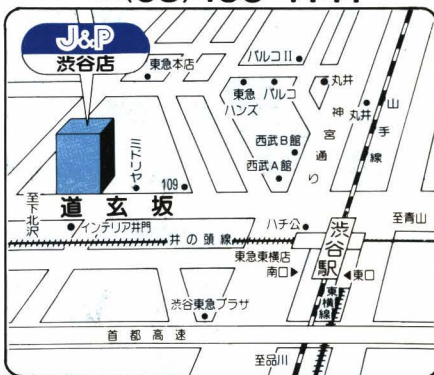
東京都渋谷区道玄坂2丁目28番4号(〒150)
☎(03)496-4141

J&Pはハードとソフトをシステムで提案します。J&Pは責任をもってサポートします。

大阪日本橋 テクノランド

大阪日本橋 メディアランド

京都寺町店

くずは店

千里中央店

名古屋 栄ノバ店

渋谷店
東京都渋谷区道玄坂2丁目28番4号(〒150)
☎(03)496-4141

大阪日本橋 テクノランド
大阪市浪速区日本橋5丁目6番7号(〒556)
☎(06)644-1413

大阪日本橋 メディアランド
大阪市浪速区日本橋4丁目11番3号(〒556)
☎(06)644-1613

京都寺町店
京都市下京区寺町通仏光寺下ル恵美須之町549(〒600)
☎(075)341-3571

くずは店
枚方市楠葉並木町2丁目2-2(〒573)
☎(0720)56-8181

千里中央店
豊中市新千里東町1-3-204千里サンタウン3F(〒565)
☎(06)834-4141

名古屋 栄ノバ店
名古屋市中区栄3丁目4-5 SAKAE NOVA 6F(〒460)
☎(052)261-9201

ここまで身近になったMZ。

●写真のカラーディスプレイMZ-1D05およびカセットレコーダ(一般用)は別売です。

いま大いなる発展性を秘めて驚異のハイコストパフォーマンス。

同じ選ぶなら初めから本モノを…将来性まで見きわめて選びたい――。いまMZがパソコンの機能と価格の概念を破りました。上達に合わせて進化する先駆のクリーン設計、家庭用カラーTVが使える、そして高度なシステムへの可能性を秘めた優れた拡張性。さらにデータレコーダ、4色カラープロッタプリンタまで本体に収納できる高密度オールインワンタイプ。MZ-700シリーズは、パソコンの在り方を徹底した追求した、まさにマニアも驚く本格仕様です。気軽に触れ合える、高度なテクニックも駆使できる。ホビーから実務まであらゆる目的に、そしてあらゆる人々に存分に活用していただきたい自信作です。

パーソナルコンピュータ

MZ-711

標準価格
79,800円

▶MZ-700シリーズとしてMZ-711の他、データレコーダ内蔵の**MZ-721標準価格89,800円**およびデータレコーダ・カラープロッタプリンタ内蔵の**MZ-731標準価格128,000円**があります。

MZ-700シリーズが収納できる
システムキャリングケース
MZ-1X04R/G標準価格19,800円

〈MZ-711の主な特長〉●高機能・高速CPU Z80A(3.6MHz)搭載●メインメモリ64KバイトRAM標準実装のクリーンメモリシステム●カラー対応BASIC装備●BASICを考慮した使いやすいキー配列●ひらがな、英小文字対応(ディスプレイ)●家庭用カラーTV、専用カラーディスプレイ(別売)による多彩なビジュアル対応●外部プリンタ用インターフェイス内蔵●MZ-80Kシリーズ・80C・1200のシステムソフト(PASCAL、マシンランゲージ等)が活用可能●グラフィック機能を装備した4色カラープロッタプリンタ内蔵可能●ディスプレイの使用できない所でも、その代用として活用できるプリンタとの対話モード装備(プリンタ要)●外形寸法・重量:440(幅)×305(奥行)×86(高さ)mm・3.6kg

シャープ株式会社 本社 〒545 大阪市阿倍野区長池町22番22号☎(06)621-1221(大代表)
●お問い合わせは…本社内国内家電営業本部システム機器営業企画部またはシャープ㈱国内産機営業本部 〒162 東京都新宿区市谷八幡町8番地☎(03)260-1161(大代表)

oh!MZ ㈱日本ソフトバンク発行 Printed in Japan **定価480円** 雑誌02179―7